NAG8-092
IN-34-CR
95P

STUDIES OF PRESSURE-VELOCITY COUPLING SCHEMES
FOR ANALYSIS OF
INCOMPRESSIBLE AND COMPRESSIBLE FLOWS

by

GREGORY WAYNE BURGREEN

(NASA-CR-181526) STUDIES OF N88-12748
PRESSURE-VELOCITY COUPLING SCHEMES FOR
ANALYSIS OF INCOMPRESSIBLE AND COMPRESSIBLE
FLOWS M.S. Thesis (Alabama Univ.) 95 p Unclas
CSCL 20D G3/34 0110651

A THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Engineering
in The Department of Mechanical Engineering
of
The School of Graduate Studies
of
The University of Alabama in Huntsville

1987

ACKNOWLEDGEMENTS

The author would like to express sincere appreciation to his co-advisors, Dr. S.T. Wu and Dr. C.P. Chen. Thanks are due to Dr. Wu for his steady guidance and support over the course of this research effort. The author owes special acknowledgement and gratitude to Dr. Chen for his helpful insight, continual encouragement, and personal interest regarding this student's progress and understanding of the research material.

The author also expresses his appreciation to Dr. G.R. Karr for serving as a committee member and for providing useful comments which increased the overall quality of this thesis.

Finally, the author is sincerely grateful to Gay Hudgins for her spiritual and emotional support throughout his graduate work.

The author gratefully acknowledges the financial support for this study provided by NASA Marshall Space Flight Center grant NGT 01-008-803, NASA/MSFC grant NAG8-092, and NASA/MSFC grant NAG8-053.

PRECEDING PAGE BLANK NOT FILMED

ABSTRACT

This study examines in detail two pressure-velocity coupling schemes, both of which solve the fully implicit discretized equations governing the flow of fluids, and assesses each one's capability of performing large Reynolds number, low Mach number compressible flow calculations. The semi-implicit iterative SIMPLE algorithm, which was originally developed for incompressible flows, is extended to handle transient compressible flow calculations. This extension takes into account a strong coupling between the pressure and temperature through a correction procedure, based on the equation of state. Results obtained from the extended SIMPLE algorithm are then compared to similar results obtained from the non-iterative PISO algorithm.

Both time-dependent and steady state calculations were performed using an axisymmetric 2:1 pipe expansion geometry and laminar flow conditions corresponding to Reynolds number of 1000 and Mach number of 0.20.

For calculations simulating a time-dependent compression/expansion process, both schemes exhibit transient features in excellent agreement with each other, and moreover, the PISO method shows a significant computational time reduction of 60% over the SIMPLE scheme, regardless of the time step size or grid size employed. The effects of

PRECEDING PAGE BLANK NOT FILMED

numerical diffusion are shown to be significant in these calculations.

For steady state compressible flows, however, the SIMPLE algorithm displays increasing computational efficiency over the PISO method as the time step sizes employed to reach steady state conditions are decreased. The accuracy of the steady state results are questionable for both algorithms when very large time step sizes are used to march to steady state conditions.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

LIST OF SYMBOLS

a	combined convection and diffusion coefficient
A	control volume interface area
A_0	central coefficient of control volume in PISO
b	partial source term
c	velocity-correction coefficient of PISO
C_p	constant-pressure specfic heat
C_v	constant-volume specific heat
d	velocity-correction coefficient of SIMPLE
dt	time step size
H	Σa_{nb} operator on argument in PISO
k	TDMA sweep number
L	length of duct
max	maximum value of argument
Ma	Mach number
p	pressure
Pr	Prandtl number
r	radial spatial coordinate
R	gas constant
Re	Reynolds number
S	full source term
t	time
T	temperature

u	axial velocity
v	radial velocity
V	volume in Chap. III, or peak velocity in Chap. V
x	axial spatial coordinate
Δ	difference
ϕ	general dependent variable
Γ	general exchange coefficient
μ	laminar viscosity
ρ	density
Σ	summation
τ	characteristic time period for steady state case

Superscripts

n	time step number
o	previous time step value
*	latest current estimate of field variable
,*	intermediate PISO field variable values
'	correction value to *

Subscripts

e	control volume's east interface
E	main node east of control volume
i	summation index
n	control volume's north interface
nb	neighboring main nodes
N	main node north of control volume
s	control volume's south interface

s	main node south of control volume
T	related to PISO's expansivity terms
P	main node of control volume
w	control volume's west interface
W	main node west of control volume
ϕ	general dependent variable

CHAPTER I INTRODUCTION

The partial differential equations governing fluid flows are nonlinear and too complex to obtain the solution in any analytical form, therefore computational methods for simulating flow fields must be employed. Since problems of practical interest include a very wide range of flow situations, it would be desirable to develop a single numerical algorithm capable of reliably handling many flow conditions in an efficient manner.

The numerical modeling of the physical mechanisms of turbulence effects, multiphase flows, and combustion are very complex individually, and the formulation of an algorithm combining all of the above mechanisms would be a formidable undertaking. However, an approach can be adopted such that one aspect of the numerical simulation can be considered individually and collectively common to the above processes; this aspect being the Navier-Stokes flow solver. Therefore, the development of a versatile and generalized flow solver which could efficiently handle both incompressible and compressible flows, i.e. flow at all speeds, in steady state or time dependent flow conditions would be the first logical step in producing a fully comprehensive fluid flow simulator.

In the early applications of numerical flow modeling [Gosman et al. (1969)], vorticity and stream function were

ORIGINAL PAGE IS
OF POOR QUALITY

usually the calculated variables. This approach has the advantage of eliminating the pressure from the governing equations, thereby eliminating the difficulties associated with the determination of pressure. However, the shortcomings of the stream function/vorticity method are numerous. This method is limited to incompressible flow. The boundary values of vorticity at the wall are difficult to specify. The extension of the stream function/vorticity method from two to three dimensions, for which a stream function does not exist, yields a very complex formulation. Hence, techniques such as this are excluded as viable candidates for a generalized flow solver due to their inherent lack of generality.

Due to the difficulty of visualizing and interpreting the meanings of vorticity and stream functions, methods based on the physically meaningful so-called primitive variables, namely pressure, density, and velocity components, gained preference. The earliest development of primitive variable schemes were the semi-implicit transient Marker-and-Cell method (MAC) [Harlow et al. (1965)] and Simplified MAC (SMAC) [Harlow et al. (1970)] by the Los Alamos group. The most widely recognized primitive variable Navier-Stokes solvers can be generally classified into two schemes: density-velocity schemes and pressure-velocity schemes.

Making use of density's explicit appearance in the continuity equation, the density-velocity schemes choose density as a main dependent variable, whereby pressure is calculated from it via an equation of state. Many density-velocity coupling schemes [Briley and McDonald (1980), Pulliam and Steger (1980)] have a disadvantage when the incompressible flow limit approaches as the linkage between pressure and density weakens in the low Mach number range. Because the variations of density in incompressible flows cease to relate to those of pressure, the density-velocity schemes are applicable to compressible flows only and, therefore, should not be used as a generalized flow solver.

Many existing methods developed specifically for incompressible flows surmount this problem by treating pressure as a primary dependent variable. These pressure-velocity coupling schemes are equally valid for compressible flows, which endow methods based on it with great versatility. These methods basically derive a working pressure equation through joint manipulation of the momentum and continuity conservation equations.

Existing methods which utilize pressure-velocity coupling fall into two catagories, namely, semi-implicit and fully implicit schemes. Because of their reliance on explicit differencing, semi-implicit techniques can be at a disadvantage for time-dependent computations, since the time-step size necessary to retain stability of such methods may drastically impair the efficiency of the algorithm particularly when applied to the calculation of steady-state flows. Implicit methods on the other hand, do not suffer from time step restrictions. They do, however, require the solution of sets of non-linear simultaneous equations for each of the dependent variables per time step.

The most popular method for solving pressure-velocity coupled incompressible flows is the SIMPLE algorithm of Patankar and Spalding (1972) and it variants: SIMPLER by Patankar (1980), SIMPLEC by Van Doormaal and Raithby (1984), SIMPLEX by Van Doormaal and Raithby (1985), and SIMPLEST in Sha (1985). The advantages gained by the implicit differencing of the SIMPLE method, which is based on a pressure correction procedure, are offset by the use of iteration which makes time dependent calculations rather expensive as iteration is required at each time step. The SIMPLE methods can be extended to handle compressible flow calculations as shown by Van Doormaal et al. (1987). This conventional compressible approach as implemented by Van Doormaal et al. (1987) accounts for additional variations in density through an equation of state-based pressure-density coupled correction scheme. Although applicable to a wide variety of flows, there are certain flow situations in which the conventional approach is inappropriate and fails to yield acceptable results. Recognized and addressed by Gosman and Watkins (1977), these flows are ones in which the temperature is strongly coupled with the pressure and velocity, such as chemically reacting flows and compression/expansion-type processes.

Another method for handling the pressure-velocity coupling of implicitly-differenced fluid flow equations is the recently introduced non-iterative PISO algorithm of Issa (1985). This method splits the process of the solution into a series of predictor and corrector steps such that, at each step, a simplified set of equations in terms of a single unknown variable is obtained. The PISO algorithm has exhibited a very efficient and robust nature when applied to a variety of flows as shown by Benodekar et al. (1985) and Issa et al. (1986).

In this study we shall extend the incompressible SIMPLE algorithm of Patankar (1980) including the time-transient and non-isothermal modifications of Bai (1986) and Bai et al. (1987) to handle compressible flow calculations. This extension will take into account the strong coupling between the pressure and temperature. Results obtained from the extended SIMPLE algorithm simulating a time dependent compression/expansion process will be compared to similar results obtained from the non-iterative PISO algorithm. Finally, steady state compressible flow calculations will be presented and compared for both algorithms.

CHAPTER II GOVERNING EQUATIONS

The governing conservation equations for mass, momentum, and thermal energy in axisymmetric coordinates with x-r-θ velocity components u-v-w can be written in a differential equation of general form [Patankar (1980)]

$$\frac{\partial}{\partial t}(\rho\phi) + \frac{\partial}{\partial x}(\rho u\phi) + \frac{1}{r}\frac{\partial}{\partial r}(r\rho v\phi) = \frac{\partial}{\partial x}\left(\Gamma_\phi\frac{\partial\phi}{\partial x}\right) + \frac{1}{r}\frac{\partial}{\partial r}\left(r\Gamma_\phi\frac{\partial\phi}{\partial r}\right) + S_\phi \tag{2.1}$$

where ρ is the fluid density and ϕ is any of the dependent variables used. The diffusion coefficients Γ_ϕ and the source terms S_ϕ in each equation for Newtonian fluid [Bird, Stewart, and Lightfoot (1960)] are listed in Table 2.1.

The pressure and density are related by an equation of state, which is taken in this study to be that of a perfect gas such that

$$p = \rho RT \tag{2.2}$$

where R is the gas constant.

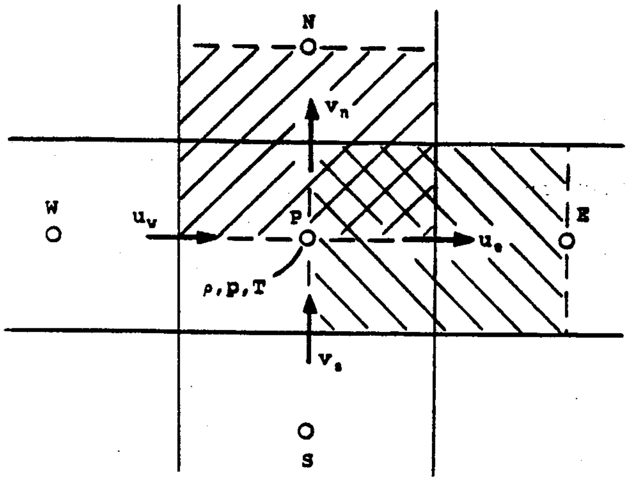
Table 2.1 Exchange Coefficients and Source Terms for ϕ

Variable ϕ	Γ_ϕ	Source S_ϕ
Continuity		
1	0	0
Momentum		
u	μ	$-\frac{\partial p}{\partial x} + \frac{1}{3}\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial x}\right) + \frac{1}{r}\frac{\partial}{\partial r}\left(r\mu\frac{\partial v}{\partial x}\right)$ $-\frac{2}{3}\frac{1}{r}\frac{\partial}{\partial x}\left(\mu v\right) - \frac{2}{3}\frac{\partial}{\partial x}\left(\mu\frac{\partial v}{\partial r}\right)$
v	μ	$-\frac{\partial p}{\partial r} + \frac{1}{3}\frac{1}{r}\frac{\partial}{\partial r}\left(r\mu\frac{\partial v}{\partial r}\right) - \frac{2}{3}\frac{\partial}{\partial r}\left(\mu\frac{\partial u}{\partial x}\right)$ $+\frac{\partial}{\partial x}\left(\mu\frac{\partial u}{\partial r}\right) - \frac{4}{3}\mu\frac{v}{r^2}$
Thermal Energy		
C_vT	$\frac{\mu C_p}{Pr}$	$-p\left(\frac{v}{r} + \frac{\partial v}{\partial r} + \frac{\partial u}{\partial x}\right)$ $+\frac{4}{3}\mu\left(\frac{\partial v}{\partial r}\right)^2 + \frac{4}{3}\mu\left(\frac{\partial u}{\partial x}\right)^2$ $+\mu\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial x}\right)^2 - \frac{4}{3}\mu\frac{\partial v}{\partial r}\frac{\partial u}{\partial x}$ $-\frac{4}{3}\mu\frac{v}{r}\left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial x}\right) + \frac{4}{3}\mu\left(\frac{v}{r}\right)^2$

CHAPTER III FINITE DIFFERENCE EQUATIONS

A. Computational Mesh

The finite difference equations are solved on the staggered grid mesh system, in which the velocity components are stored at locations staggered with respect to those at which the pressures, densities, and temperatures are stored. Figure 3.1 shows such a staggered arrangement.



□ Mass and Energy Conservation Control Volume

▧ u-velocity Control Volume

▨ v-velocity Control Volume

○ Density, Pressure, and Temperature Nodes

→ u-velocity Node

↑ v-velocity Node

Figure 3.1 Staggered Grid Arrangement of Nodes and Control Volumes

The staggered grid has a number of advantages [Ferziger (1987)]. First, the differencing scheme for mass, momentum, and energy conservation can be developed in a more realistic and natural way when the staggered grid approach is employed. The mass flux terms based on the staggered velocities are well suited to the difference form of the mass conservation equation, and momentum's pressure gradient terms are centered about the velocities they drive, thus avoiding the "checkerboard effect" in the pressure field [Patankar (1980)]. Also, treatment of the boundary conditions is easier when the staggered grid system is used, especially for the pressure. With staggered grids, the domain boundary can be chosen to fall on velocity nodes, thereby making it possible to avoid specifying any explicit boundary conditions for the pressure at the surface. The major disadvantage in using the staggered grid approach is that separate control volumes must to be constructed for each variable, which requires more memory than using a non-staggered grid approach.

B. Discretization

The discretized transport equations are obtained using a finite-volume technique in which the difference equations are formally derived by integrating the differential conservation equations over its respective control volume.

The unsteady terms of the general differential equation (1) are handled using a first order accurate backward temporal difference scheme, although any second order differencing scheme can be easily incorporated.

The convective and diffusive terms are represented by the hybrid difference scheme, which is thoroughly presented by Patankar (1980). The basic premise of the hybrid scheme is that second order accurate central differencing of the convective term is preferred when the cell Reynolds number (Peclet number), Pe = $\rho u \Delta x/\Gamma$, is less than two. For larger values of Pe, first order accurate upwind differencing of the convective term is required to obtain stable solutions for highly convective flows. The hybrid approach can be briefly summarized by

$$\left[\rho_e u_e \phi_E - \rho_e u_e \phi_P - \Gamma_e \left(\frac{\partial \phi}{\partial x}\right)_e \right] = \left\{ \max\left(-\rho_e u_e, \ \frac{\Gamma_e}{\Delta x} - \frac{\rho_e u_e}{2}, \ 0 \right) \right\} (\phi_P - \phi_E) \tag{3.1}$$

where Δx is the grid distance between adjacent nodes.

The deficiencies of the hybrid approach are the upwind scheme's first order accuracy and a form of discretization error often referred to as "numerical diffusion" [Patankar (1980)]. Higher order hybrid approaches can be adopted which replaces the upwind differencing by either the QUICK scheme of Leonard (1977) or the skewed upwind method of Raithby (1976) for high cell Reynolds numbers. Both the QUICK and skewed upwind methods are more accurate than the simple upwind method on a given grid, but they sometimes produce solutions with oscillations [Syed et al. (1985)].

The final form of the fully implicit discretized general differential equation (2.1) is obtained as presented by Patankar (1980)

$$\left(a_P + \rho^0 \frac{\Delta V}{\Delta t}\right) \phi_P = \sum a_{nb}\phi_{nb} + S_\phi \Delta V + \rho^0 \frac{\Delta V}{\Delta t} \phi_P^0 \qquad (3.2)$$

where the coefficient "a" represents the finite-difference representation of the spatial convective and diffusive fluxes associated with ϕ, ΔV is the volume of the appropriate control volume, and Δt is the time step size. In conformity with the fully implicit practice, the values superscripted with ^{0} are to be regarded as those values existing at the "old" time step, while all values with no superscript are regarded as "new" time-step values.

C. Solution of the Algebraic Equations

By virtue of the simple form of the discretized equation (3.2), the solution for the dependent field variable can be conveinently obtained by an algebraic equation solver using the TDMA (Tri-Diagonal Matrix Algorithm).

Although equation (3.2) is cast and solved as a linear algebraic equation, nonlinear situations arise when the momentum and energy conservation equations are represented in the form of (3.2). This nonlinearity exists when the coefficients a_P and a_{nb} themselves depend on ϕ, or the source term S_ϕ is a nonlinear function of ϕ. Iterative methods of solution must then be employed to handle the nonlinearity by solving the set of equations repeatedly with updated coefficients or source term values evaluated from the newly obtained ϕ values. In this way, the degree of accuracy to which the nonlinearity of the equation is handled depends upon the number of times the TDMA is repeatedly solved. In this study, a line-by-line iterative scheme is used in conjunction with the TDMA in order to solve the set of fully implicit discretized equations (3.2). Since boundary condition information is transmitted at once into the interior of the calculation domain, the line-by-line method possesses faster convergence rates than point-by-point methods.

CHAPTER IV SOLUTION PROCEDURES

Because of the roles the pressure and its gradients play in each of the transport equations, the determination of an accurate pressure field is essential for the complete calculation of a flow field. However, the lack of an explicit equation governing the pressure is a primary obstacle to solution of the Navier-Stokes equations. In the following sections, methods of predicting the pressure field, which use the Poisson equation for the pressure in place of the continuity equation, will be presented. The extension of these methods into generalized methods capable of handling incompressible as well as compressible flows will be developed.

A. SIMPLE Algorithm

1. Incompressible Flows

Examining the governing u_e momentum equation of the form given by equation (3.2) with the general variable ϕ equated to u_e and the variables Γ_ϕ and S_ϕ specified in Table 2.1, the resulting discretized equation becomes:

$$\left(a_e + \rho^0\frac{\Delta V}{\Delta t}\right) u_e = \sum a_{nb}u_{nb} + (p_P - p_E)\, A_e r_e + b + \rho^0\frac{\Delta V}{\Delta t}\, u_e^0 \tag{4.1}$$

where b is the remaining source terms after the pressure gradient is explicitly written and r_e is the radius locating the center of the face area, A_e, of the u-velocity control volume from the cylindrical coordinate line of symmetry. Note that for incompressible flows, density is considered constant (i.e. "old" densities equal "new" densities). Rewriting equation (4.1) in terms of estimated values of velocities u* and pressures p* and subtracting the resulting equation from (4.1) gives fully implicit velocity correction equations, given by

$$\left(a_e + \rho^0 \frac{\Delta V}{\Delta t}\right) u'_e = \sum a_{nb} u'_{nb} + (p'_P - p'_E)\, A_e r_e \qquad (4.2)$$

where

$$u' = u - u^* \qquad \text{or} \qquad u = u^* + u' \qquad (4.3a,b)$$

$$p' = p - p^* \qquad \text{or} \qquad p = p^* + p' \qquad (4.4a,b)$$

Similar equations apply to the v-velocity field.

Unless the correct pressure field is used in equation (4.1), the resulting velocities will not satisfy continuity. One way of improving the solutions of equation (4.1) is to correct the pressure field p via equation (4.4b), thus requiring an equation for obtaining pressure-correction p'.

Characteristic of the SIMPLE method, the term $\Sigma a_{nb} u'_{nb}$ in equation (4.2) is neglected [Patankar (1980)] thus, velocity-correction in terms of pressure-correction only is obtained, given by

$$u_e' = d_e(p_P' - p_E') \qquad (4.5)$$

where

$$d_e = \frac{A_e r_e}{\left(a_e + \rho^0 \frac{\Delta V}{\Delta t}\right)} \qquad (4.6)$$

Substituting equation (4.5) into equation (4.3b) gives a linearized velocity equation for u_e, which in turn is substituted into discretized continuity and gives the equation in terms of pressure-correction p' only. Once the p' equation has been solved, an updated velocity field can be obtained from equations (4.5) and (4.3b).

The above outlined pressure-correction procedure has been termed the SIMPLE (<u>S</u>emi-<u>I</u>mplicit <u>M</u>ethod for <u>P</u>ressure <u>L</u>inked <u>E</u>quations), where the semi-implicitness is derived from the fact that the $\Sigma a_{nb}u'_{nb}$ was neglected in (4.2). The SIMPLE algorithm requires an iterative-type solution procedure due to:

i). the inter-equation coupling between the transport equations and

ii). the nature of the correction-type process of equations (4.3b) and (4.4b).

A variant of the SIMPLE method is the SIMPLEC approximation described by Van Doormaal et al. (1984), which in general produces a faster rate of convergence for certain flow situations. The SIMPLE algorithm neglects the $\Sigma a_{nb}u'_{nb}$ term in equation (4.2), but a term of similar magnitude is retained in a_e on the left-hand side of the equation. SIMPLEC introduces a "consistant" approximation, such that $\Sigma a_{nb}u'_e$ is subtracted from both sides of equation (4.2) giving

$$\left(a_e + \rho^0\frac{\Delta V}{\Delta t} - \Sigma a_{nb}\right) u'_e = \sum a_{nb}(u'_{nb} - u'_e)$$
$$+ (p'_P - p'_E) A_e r_e \qquad (4.7)$$

Now in the SIMPLEC approximation, the $\Sigma a_{nb}(u'_{nb} - u'_e)$ term is neglected and the velocity correction is given by equation (4.5), except now

$$d_e = \frac{A_e r_e}{\left(a_e + \rho^0\frac{\Delta V}{\Delta t} - \Sigma a_{nb}\right)} \qquad (4.8)$$

The role of pressure in the incompressible flow case is to correct velocities through the momentum equations such that the resulting velocities conserve mass. This interpretation of the role of pressure arises because only velocity and not pressure appears in the mass conservation equation [Van Doormaal et al. (1987)].

2. Compressible Flows

Due to the nature of compressible flows, additional variations in the density and temperature must be considered. The conventional approach for treating compressible flows will be presented and a generalized approach applicable to a wide variety of flow situations, for which the conventional approach suffers, will be developed.

a. Conventional Approach

To account for the spatial variation of the density field in compressible flows, the SIMPLE algorithm's pressure-correction procedure for computing incompressible flows is extended by Van Doormaal et al. (1987) such that an updated density field can be conveniently obtained from a pressure-correction p'. The density-correction formula is obtained as

$$\rho = \rho^* + \left(\frac{\partial \rho}{\partial p}\right) p' \qquad (4.9)$$

where ρ^* is the latest current estimate of density. The compressibility correction factor $(\partial \rho/\partial p)$ is calculated from the equation of state. For a perfect gas, $(\partial \rho/\partial p) = 1/RT$.

In order to remain general, the conservation of mass equation should be satisfied by mass fluxes which include the influences of velocity correction obtained from equations (4.3b) and (4.5) and also density corrections of equation (4.9). The necessary mass flux term possessing both of these influences can be linearized by

$$(\rho u A) \simeq [\rho u^* A] + \{\rho^* u A\} - (\rho^* u^* A) \qquad (4.10)$$

where the superscript * denotes the latest current estimate of the dependent variable. The mass flux term enclosed in the [] brackets indicates that its density is approximated by the density correction formula (4.9), and the mass flux term enclosed by the {} brackets has its velocity approximated by the linearized velocity correction given by substitution of equation (4.5) into (4.3b).

The values for density in the mass flux terms (4.10) of the continuity equation are evaluated using the upwind scheme, as opposed to the practice of using averaged values of density as is utilized in the remaining conservation equations. The upwind scheme is discussed by Patankar (1980) and can be represented

$$(\rho uA)_e = \rho_P \max\{ u_e, 0\}A_e - \rho_E \max\{-u_e, 0\}A_e \qquad (4.11)$$

Substitution of the upwinded-density form of equation (4.10) into the discretized continuity equation yields a pressure-correction equation. The solution of the pressure-correction equation is used to calculate updated values of density and velocity.

The role of pressure in compressible flows becomes that the pressure must influence the velocity through the momentum conservation and density through the equation of state such that together the resulting velocities and resulting densities conserve mass.

b. Generalized Approach

It is the objective of this study to obtain a generalized scheme for calculating flow fields involving pressure-velocity coupled variables. To do so care must be taken to ensure that calculation procedures are applicable to incompressible as well as compressible flows and also that the intricate coupling of pressure-velocity-density-temperature is handled in a complete manner as possible.

If the method used for approximating the mass flux terms given by equation (4.10) is examined with the concept of generalization in mind, several observations become clear. The mass conservation equation resulting from the insertion of the linearized mass flux terms (4.10) has the attribute of being linearized in terms of both velocity and density, which ensures applicability to both incompressible as well as compressible flows. Van Doormaal et al. (1987) notes that neglecting the influence of density correction in mass conservation restricts the application of the resulting solution method to low Mach numbers and incompressible flows. Also, procedures accounting for density correction alone, with no velocity correction, limits the applications to compressible flows with very small time steps required in the low Mach number range to maintain stability.

In order for a generalized approach to handle the strong pressure-temperature coupling demanded by, for example, chemically reacting flows, buoyancy-driven flows, and expansion/compression-type processes, the conventional density correction formula (4.9) is modified such that a stronger influence from the temperature field is felt throughout the updated density field and is given by

$$\rho = \rho^{*} + \left(\frac{\partial \rho}{\partial p}\right) p' + \left(\frac{\partial \rho}{\partial T}\right) T' \qquad (4.12)$$

where $(\partial \rho/\partial T)$ is an expansivity correction factor and is calculated from the equation of state. For a perfect gas, $(\partial \rho/\partial T) = -p/RT^2$. T' is the temperature correction and in this study is simply defined as the temperature difference between successive time steps, n+1 and n, given by

$$T' = T^{n+1} - T^{n} \qquad (4.13)$$

By replacing the conventional density correction formula with the modified form (4.12), a pressure correction equation can be derived as before except that a stronger influence from the temperature field is included. The resulting generalized pressure-correction equation is given by

$$a_P p'_P = a_E p'_E + a_W p'_W + a_N p'_N + a_S p'_S + b + b_T \qquad (4.14)$$

where

$$a_E = \overset{*}{\rho}_e d_e A_e r_e + \left(\frac{\partial \rho}{\partial p}\right)_E A_e r_e \max\{-\overset{*}{u}_e,\ 0\} \qquad (4.15a)$$

$$a_W = \overset{*}{\rho}_w d_w A_w r_w + \left(\frac{\partial \rho}{\partial p}\right)_W A_w r_w \max\{\ \overset{*}{u}_w,\ 0\} \qquad (4.15b)$$

$$a_N = \overset{*}{\rho}_n d_n A_n r_n + \left(\frac{\partial \rho}{\partial p}\right)_N A_n r_n \max\{-\overset{*}{u}_n,\ 0\} \qquad (4.15c)$$

$$a_S = \overset{*}{\rho}_s d_s A_s r_s + \left(\frac{\partial \rho}{\partial p}\right)_S A_s r_s \max\{\ \overset{*}{u}_s,\ 0\} \qquad (4.15d)$$

$$\begin{aligned} a_P = {} & \left(\frac{\partial \rho}{\partial p}\right)_P \frac{\Delta V}{\Delta t} + \overset{*}{\rho}_e d_e A_e r_e + \overset{*}{\rho}_w d_w A_w r_w + \overset{*}{\rho}_n d_n A_n r_n + \overset{*}{\rho}_s d_s A_s r_s \\ & + \left(\frac{\partial \rho}{\partial p}\right)_P \Big[A_e r_e \max\{\ \overset{*}{u}_e,\ 0\} + A_w r_w \max\{-\overset{*}{u}_w,\ 0\} \\ & \qquad + A_n r_n \max\{\ \overset{*}{u}_n,\ 0\} + A_s r_s \max\{-\overset{*}{u}_s,\ 0\} \Big] \end{aligned} \qquad (4.15e)$$

$$\begin{aligned} b = {} & \frac{\Delta V}{\Delta t} \{ \overset{o}{\rho}_P - \overset{*}{\rho}_P \} - \overset{*}{\rho}_P A_e r_e \max\{\ \overset{*}{u}_e,\ 0\} + \overset{*}{\rho}_E A_e r_e \max\{-\overset{*}{u}_e,\ 0\} \\ & + \overset{*}{\rho}_W A_w r_w \max\{\ \overset{*}{u}_w,\ 0\} - \overset{*}{\rho}_P A_w r_w \max\{-\overset{*}{u}_w,\ 0\} \\ & - \overset{*}{\rho}_P A_n r_n \max\{\ \overset{*}{u}_n,\ 0\} + \overset{*}{\rho}_N A_n r_n \max\{-\overset{*}{u}_n,\ 0\} \\ & + \overset{*}{\rho}_S A_s r_s \max\{\ \overset{*}{u}_s,\ 0\} - \overset{*}{\rho}_P A_s r_s \max\{-\overset{*}{u}_s,\ 0\} \end{aligned} \qquad (4.15f)$$

$$b_T = \frac{\Delta V}{\Delta t}\left(\frac{\partial \rho}{\partial T}\right)_P T'_P$$
$$+ \left(\frac{\partial \rho}{\partial T}\right)_P T'_P A_e r_e \max\{\ u_e^*,\ 0\} - \left(\frac{\partial \rho}{\partial T}\right)_E T'_E A_e r_e \max\{-u_e^*,\ 0\}$$
$$+ \left(\frac{\partial \rho}{\partial T}\right)_P T'_P A_w r_w \max\{-u_w^*,\ 0\} - \left(\frac{\partial \rho}{\partial T}\right)_W T'_W A_w r_w \max\{\ u_w^*,\ 0\}$$
$$+ \left(\frac{\partial \rho}{\partial T}\right)_P T'_P A_n r_n \max\{\ u_n^*,\ 0\} - \left(\frac{\partial \rho}{\partial T}\right)_N T'_N A_n r_n \max\{-u_n^*,\ 0\}$$
$$+ \left(\frac{\partial \rho}{\partial T}\right)_P T'_P A_s r_s \max\{-u_s^*,\ 0\} - \left(\frac{\partial \rho}{\partial T}\right)_S T'_S A_s r_s \max\{\ u_s^*,\ 0\} \qquad (4.15g)$$

where $\rho_e{}^*$, $\rho_w{}^*$, $\rho_n{}^*$, and $\rho_s{}^*$ correspond to the upwind nodal values of density, and d_w, d_n, and d_s are similar to d_e as defined in equation (4.6). Notice that the above pressure-correction equation incorporated:

i). an upwind-like nature as defined by (4.11),

ii). influences from the compressibility correction term of (4.9), and

iii). the b_T source term, which contains the expansivity correction term influences of equation (4.12). Upon solving (4.14) for the p' field, updated density fields are calculated from equation (4.12) and updated velocity fields are calculated from equations (4.3b) and (4.5). Each of these fields are now sufficiently coupled to temperature.

The generalized SIMPLE algorithm consists of the following steps:

1. Guess the pressure p^*, density ρ^*, and temperature T^*.

2. Solve the momentum equation (4.1) based on p^*, ρ^*, and T^* for u^* and v^* velocity fields.
3. Solve the p' equation (4.14) and update the pressure by equation (4.4b).
4. Update density using the density correction formula (4.12).
5. Update velocities u and v by the velocity correction equations (4.3b) and (4.5)
6. Update the temperature field using equation (3.2) and appropriate values for the general variable ϕ.
7. Repeat steps 2 through 6 until convergence is reached.
8. March to next time step in the time-marching formulation (or assume that steady-state conditions have been achieved in the iterative formulation.)

The above mentioned methodology requires an iterative procedure using the TDMA equation solver. The use of iteration for time dependent calculations are expensive since iteration is required at each time step. Also, convergence of the solution domain is troublesome and can not always be assured.

Recently, non-iterative methods and fractional step methods [Kim and Moin (1980)] have been developed which makes use of splitting methods in order to efficiently obtain flow field solutions that satisfy the conservation equations without resorting to iteration. One such non-iterative method is the PISO (Pressure-Implicit with Splitting of Operators) algorithm of Issa (1985) which is capable of handling the pressure-velocity coupling of implicitly discretized fluid flow equations and is described below.

B. PISO Algorithm

1. Compressible Flows

The discretized governing equations used in the PISO algorithm are developed using the same procedures outlined in Chapter II. The final form of the general discretized differential equation is, however, modified using the continuity equation such that the following form is obtained.

$$\left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho}\right) \rho \phi_P = \sum a_{nb}\phi_{nb} + S_P \Delta V + \rho^0 \frac{\Delta V}{\Delta t} \phi_P^0 \qquad (4.16)$$

where $A_0 = a_p + \rho_e u_e A_e r_e - \rho_w u_w A_w r_w + \rho_n u_n A_n r_n - \rho_s u_s A_s r_s$ (Issa 1986b) with the same a_p used in equation (3.2). The primary difference between equations (4.16) and (3.2) is that (4.16) has the left hand side in terms of the "new" density, which is more amenable to the predictor-corrector methodology of PISO.

The nonlinearity in equation (4.16) arising from the dependency of the coefficients A_0 and a_{nb} on the field variable ϕ, which is normally treated through iteration in the TDMA line-by-line solver, is now handled by treating the coefficients as constants evaluated at the "old" time level ϕ values.

The formal order of accuracy of the approximations of the exact solution obtained by the splitting of operations depend on the number of operation-splittings used. It has been shown by Issa (1985) that at least two corrector stages will resolve a pressure field free from the influences of error in the divergence of the calculated velocities. This two stage splitting principle as applied to pressure-velocity coupled variables is now presented below.

Let the field variables of the converged solution at the "old" time step be represented by the superscript ^{0} and the values of velocity and temperature at the predictor, first corrector, and second corrector steps be denoted by superscripts *, **, and ***, respectively. The corresponding intermediate values of the remaining field variables (p, ρ) will also be denoted by the superscripts *, **, and ***.

1). Momentum Predictor Step

Substituting the pressure and density fields prevailing at the "old" time step into equation (4.16), the predictor velocity field $u_e{}^*$ may be obtained by solving

$$\left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_e{}^0}\right) \rho_e{}^0 u_e{}^* = H(u_e{}^*) + (p_P^0 - p_E^0) A_e r_e + S_e \Delta V + \rho_e^0 \frac{\Delta V}{\Delta t} u_e^0 \tag{4.17}$$

where the $H(u_e{}^*)$ operator stands for $\Sigma a_{nb} u_{nb}{}^*$. It should be noted that at this stage ρ^0, p^0, and u^* satisfy momentum conservation, but continuity is not satisfied by ρ^0 and u^*.

2). <u>First Momentum Corrector Step</u>

The momentum equation is now written in terms of a new velocity field u_e^{**} and its corresponding new pressure field p^* and new density field ρ^* giving

$$\left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_e{}^0}\right) \rho_e{}^* u_e{}^{**} = H(u_e{}^*) + (p_P^* - p_E^*)A_e r_e + S_e \Delta V + \rho_e^0 \frac{\Delta V}{\Delta t} u_e^0$$

(4.18)

An incremental form of the momentum equation is obtained by subtracting equation (4.17) from (4.18) giving

$$\rho_e{}^* u_e{}^{**} - \rho_e{}^0 u_e{}^* = -\left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_e{}^0}\right)^{-1} \left[(p_E^* - p_E^0) - (p_P^* - p_P^0) \right] A_e r_e$$

$$= - c_e \; [(p^* - p^0)_E - (p^* - p^0)_P] \; A_e r_e \qquad (4.19)$$

where

$$c_i = A_i r_i \left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_i{}^0}\right)^{-1} \qquad (i = e,w,n,s) \qquad (4.20)$$

Substituting equation (4.19) and its similarly derived west, north, and south counterparts into the continuity equation based on intermediate values of velocity and density, which is given by

$$\frac{\Delta V}{\Delta t} (\rho_P{}^* - \rho_P{}^0) + (\rho^* u^{**})_e A_e r_e - (\rho^* u^{**})_w A_w r_w$$
$$+ (\rho^* u^{**})_n A_n r_n - (\rho^* u^{**})_s A_s r_s = 0 \qquad (4.21)$$

leads to the incremental pressure equation written as

$$c_e[(p^* - p^0)_E - (p^* - p^0)_P] A_e r_e - c_w[(p^* - p^0)_P - (p^* - p^0)_W] A_w r_w$$
$$+ c_n[(p^* - p^0)_N - (p^* - p^0)_P] A_n r_n - c_s[(p^* - p^0)_P - (p^* - p^0)_S] A_s r_s$$
$$= \frac{\Delta V}{\Delta t} (\rho_P{}^* - \rho_P{}^0) + (\rho^0 u^*)_e A_e r_e - (\rho^0 u^*)_w A_w r_w$$
$$+ (\rho^0 u^*)_n A_n r_n - (\rho^0 u^*)_s A_s r_s \qquad (4.22)$$

Eliminating $\rho_P{}^*$ in favor $p_P{}^*$ is accomplished by invoking the first corrector step equation of state, which is given as

$$\rho^* = p^*/RT^0 \tag{4.23}$$

The following pressure-increment equation is then obtained

$$a_P(p^*-p^0)_P = a_E(p^*-p^0)_E + a_W(p^*-p^0)_W + a_N(p^*-p^0)_N + a_S(p^*-p^0)_S + b \tag{4.24}$$

where

$$a_E = c_e A_e r_e \qquad a_W = c_w A_w r_w \tag{4.25a,b}$$

$$a_N = c_n A_n r_n \qquad a_S = c_s A_s r_s \tag{4.25c,d}$$

$$a_P = a_E + a_W + a_N + a_S + \frac{\Delta V}{\Delta t}\left(\frac{1}{RT^0}\right) \tag{4.25e}$$

$$b = (\rho^0 u^*)_w A_w r_w - (\rho^0 u^*)_e A_e r_e + (\rho^0 u^*)_s A_s r_s - (\rho^0 u^*)_n A_n r_n \tag{4.25f}$$

This equation can be solved for the (p^*-p^0) field, which yields the p^* field. Equations (4.23) and (4.19) may then be used to determine the ρ^* and u^{**} fields, respectively. The continuity relation (4.21) is now satisfied by ρ^* and u^{**}, however the momentum equation (4.18) is not satisfied to an acceptably accurate degree.

3). <u>Temperature</u> <u>Predictor</u> <u>Step</u>

After substitution of the latest values of the field variables into the thermal energy equation, the predictor temperature field T^* may be calculated from

$$\left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_P{}^*}\right) \rho_P{}^* C_v T_P{}^* = \sum a_{nb} T_{nb}{}^* + p_P{}^*(u_w{}^{**} A_w r_w - u_e{}^{**} A_e r_e + u_s{}^{**} A_s r_s - u_n{}^{**} A_n r_n) + b + \rho_P{}^0 \frac{\Delta V}{\Delta t} C_v T_P{}^0 \qquad (4.26)$$

4). <u>Second</u> <u>Momentum</u> <u>Corrector</u> <u>Step</u>

The most current updated field variables are utilized in writing the operative second corrector momentum equation, which is taken as

$$\left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_e{}^*}\right) \rho_e{}^{**} u_e{}^{***} = H(u_e{}^{**}) + (p_P{}^{**} - p_E{}^{**}) A_e r_e + S_e \Delta V + \rho_e{}^0 \frac{\Delta V}{\Delta t} u_e{}^0 \qquad (4.27)$$

Subtracting equation (4.18) from (4.27) yields, after some algebra, the incremental form of the momentum equation given by

$$\rho_e{}^{**} u_e{}^{***} - \rho_e{}^* u_e{}^{**} = \left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_e{}^*}\right)^{-1} \left\{ H(u_e{}^{**} - u_e{}^*) - A_e r_e [(p^{**}-p^*)_E - (p^{**}-p^*)_P] + A_0 \left(\frac{\rho_e{}^* - \rho_e{}^0}{\rho_e{}^0}\right) u_e{}^{**} \right\} \qquad (4.28)$$

Substituting equation (4.28) and its similarly derived west, north, and south counterparts into the continuity equation based on u^{***} and p^{**}, a pressure equation is obtained in terms of the pressure increment $(p^{**}-p^*)$.

Like the first corrector pressure equation, the remaining ρ^{**} terms originating from the continuity equation are eliminated by invoking the equation of state

$$\rho^{**} = p^{**}/RT^{*} \tag{4.29}$$

The second corrector pressure-increment equation then is written

$$a_P(p^{**}-p^{*})_P = a_E(p^{**}-p^{*})_E + a_W(p^{**}-p^{*})_W + a_N(p^{**}-p^{*})_N + a_S(p^{**}-p^{*})_S + b \tag{4.30}$$

where

$$a_E = C_e^{*}A_e r_e \qquad a_W = C_w^{*}A_w r_w \tag{4.31a,b}$$

$$a_N = C_n^{*}A_n r_n \qquad a_S = C_s^{*}A_s r_s \tag{4.31c,d}$$

$$a_P = a_E + a_W + a_N + a_S + \frac{\Delta V}{\Delta t}\left(\frac{1}{RT^{*}}\right) \tag{4.31e}$$

$$\begin{aligned} b = {} & C_w^{*}\left[H(u_w{}^{**}-u_w{}^{*}) + \left\{A_0\left(\frac{\rho^{*}-\rho^{0}}{\rho^{0}}\right)u^{**}\right\}_w\right] \\ & - C_e^{*}\left[H(u_e{}^{**}-u_e{}^{*}) + \left\{A_0\left(\frac{\rho^{*}-\rho^{0}}{\rho^{0}}\right)u^{**}\right\}_e\right] \\ & + C_s^{*}\left[H(u_s{}^{**}-u_s{}^{*}) + \left\{A_0\left(\frac{\rho^{*}-\rho^{0}}{\rho^{0}}\right)u^{**}\right\}_s\right] \\ & - C_n^{*}\left[H(u_n{}^{**}-u_n{}^{*}) + \left\{A_0\left(\frac{\rho^{*}-\rho^{0}}{\rho^{0}}\right)u^{**}\right\}_n\right] \\ & - \frac{\Delta V}{\Delta t}\,\frac{p_P^{*}}{R}\left(\frac{1}{T_P{}^{*}} - \frac{1}{T_P{}^{0}}\right) \end{aligned} \tag{4.31f}$$

$$C_i{}^{*} = A_i r_i \left(\frac{\Delta V}{\Delta t} + \frac{A_0}{\rho_i{}^{*}}\right)^{-1} \qquad (i = e,w,n,s) \tag{4.31g}$$

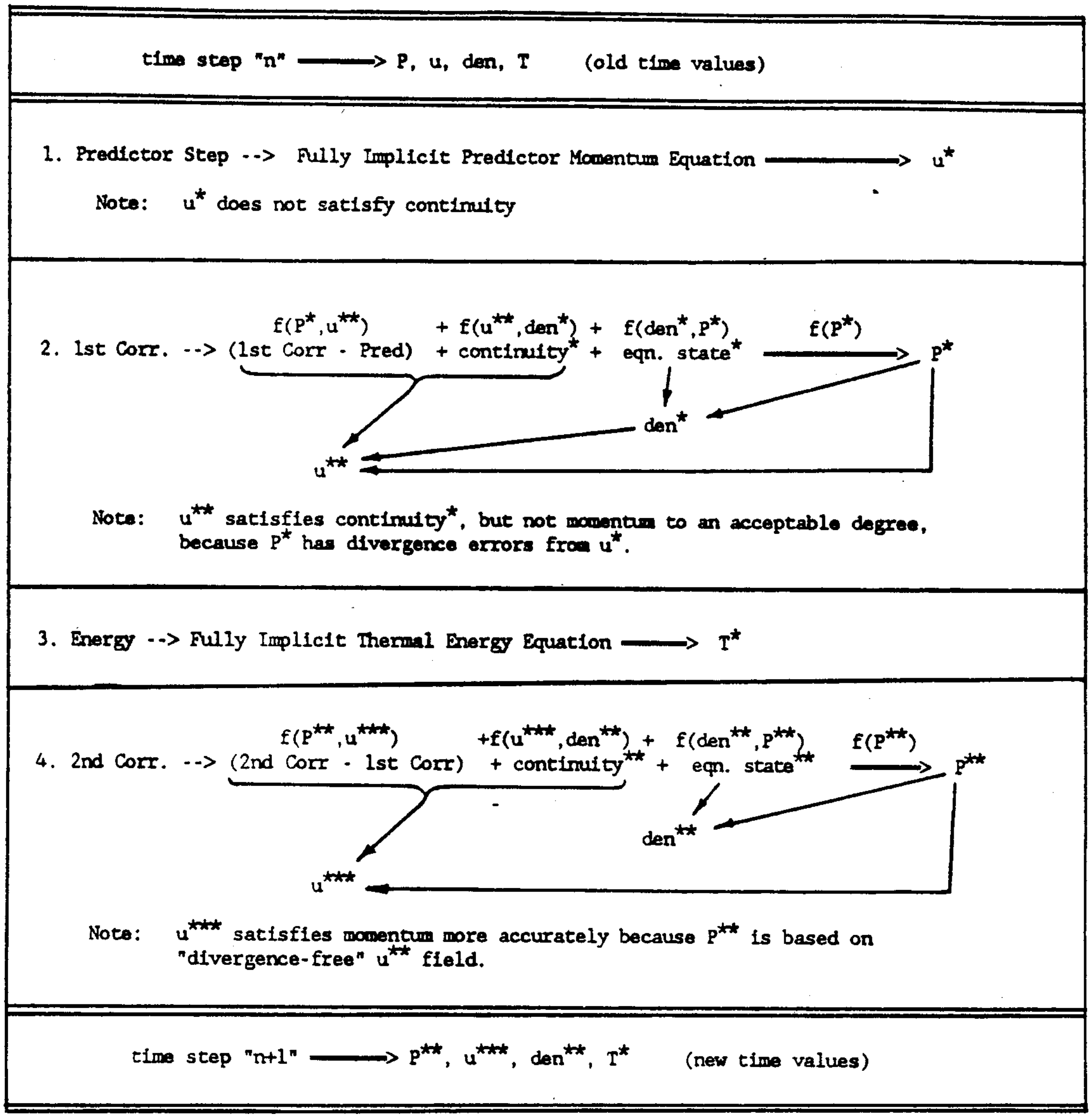
Upon solving equation (4.30) for the $(p^{**}-p^{*})$ incremental pressure field, the updated density ρ^{**} may be obtained via (4.29) and the u^{***} field can be explicitly calculated from (4.28).

Figure 4.1 illustrates in a simplified manner the non-iterative methodology adopted by the PISO algorithm for calculating the field variables through one time step. The functionally dependent variables of each corrector step operation is shown, and also the manner in which the variables are then calculated is illustrated.

2. Incompressible Flows

The PISO algorithm is equally applicable to incompressible flows. At low Mach numbers, the density becomes weakly linked to pressure, and the equation of state (2.2) ceases to apply. The derivation of the time-dependent forms of the incompressible predictor and corrector equations follow in the same manner as before, except that density is now treated as constant. For isothermal flows, the predictor step for energy conservation does not have to be solved.

ORIGINAL PAGE IS
OF POOR QUALITY


time step "n" ——> P, u, den, T (old time values)
1. Predictor Step --> Fully Implicit Predictor Momentum Equation ——> u^*
2. 1st Corr. --> $f(P^*,u^{**})$ (1st Corr - Pred) + $f(u^{**},den^*)$ + continuity^* + $f(den^*,P^*)$ eqn. state^* $\xrightarrow{f(P^*)}$ P^*
den^*
u^{**}
3. Energy --> Fully Implicit Thermal Energy Equation ——> T^*
4. 2nd Corr. --> $f(P^{**},u^{***})$ (2nd Corr - 1st Corr) $+f(u^{***},den^{**})$ + continuity^{**} + $f(den^{**},P^{**})$ eqn. state^{**} $\xrightarrow{f(P^{**})}$ P^{**}
den^{**}
u^{***}
time step "n+1" ——> P^{**}, u^{***}, den^{**}, T^* (new time values)

Note: u^* does not satisfy continuity

Note: u^{**} satisfies continuity*, but not momentum to an acceptable degree, because P^* has divergence errors from u^*.

Note: u^{***} satisfies momentum more accurately because P^{**} is based on "divergence-free" u^{**} field.

Figure 4.1 Simplified PISO Algorithm Methodology

CHAPTER V DEMONSTRATION PROBLEM AND PROCEDURES

The demonstration problem chosen in this study is characterized by large Reynolds number, low Mach number flow conditions; no attempt has been made in this study to calculate flows in the transonic or supersonic regimes. Because large Reynolds number, low Mach number-type flows occur frequently in practical situations, this problem may serve as a good benchmark problem for evaluating the performance of different fluid flow algorithms. Great versatility is required by an algorithm in the computation of large Reynolds number subsonic flows, particularly when the flow attains different speeds either in different regions or at different instants (e.g., the flow in a reciprocating engine at different parts of its cycle.) In the calculation of the low Mach number flows of practical interest, such as the elevated temperature (high speed of sound) internal flows of liquid rocket engines, the importance of using pressure-velocity coupled schemes is recognized because of the small density gradients in the low Mach number range.

A. Time-dependent Compressible Flow

1. Demonstration Problem Geometry

The time-dependent compressible flow case chosen is that of an axisymmetric laminar flow compression/expansion process in a duct with a sudden expansion at inlet and the downstream end closed, so that the fluid mass inside the duct varies with time. Figure 5.1 illustrates the test geometry used. The ratio of the duct diameter to that at inlet is 2:1 and the length of the duct is 4 times the duct diameter. The velocity profile at inlet is assumed to be uniform in the radial direction and is taken to vary sinusoidally with time, as illustrated by Figure 5.2, such that peak velocity V corresponds to a Reynolds number, based on peak velocity V and the duct diameter, of 1000 and a Mach number of 0.2. The period of the sinusoidal cycle is taken as twice the time required for a particle traveling at peak velocity V to traverse the length of the duct. The time-dependent compressible flow calculations are carried out over a full cycle of inlet velocity variations starting from rest. The density and temperature at inlet are assumed to be uniform and constant throughout the complete cycle.

Two locations within the geometry are monitored such that transient values of velocity (normalized by peak velocity V) and pressure (normalized by inlet pressure) may be analyzed during the compression/expansion cycle. The first point is located at one-sixth the distance down the duct from inlet and at 80% of the duct radius. This point is referred to as the recirculation zone, as some of the highest reverse velocities occur at this location. The second point, appropriately referred to as the centerline zone, is located on the geometry's centerline halfway down the duct.

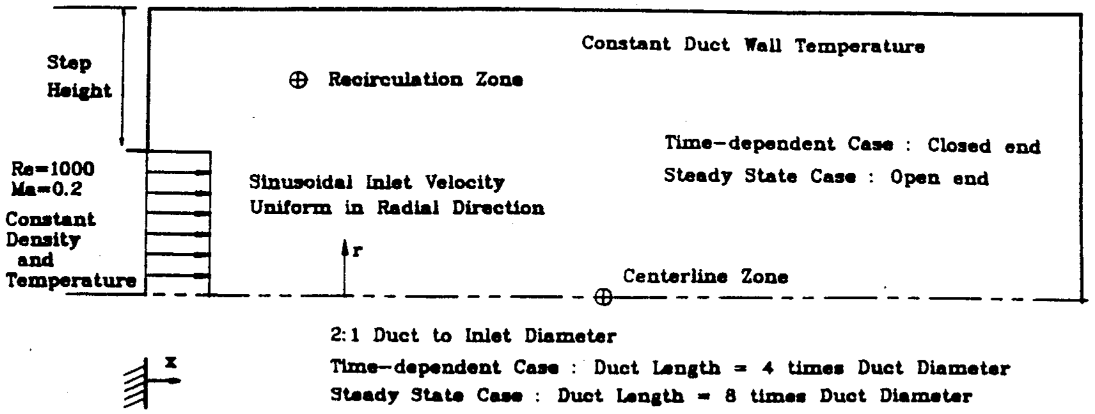

Step
Height
Constant Duct Wall Temperature
⊕ Recirculation Zone
Time-dependent Case : Closed end
Steady State Case : Open end
Re=1000
Ma=0.2
Constant
Density
and
Temperature
Sinusoidal Inlet Velocity
Uniform in Radial Direction
r
Centerline Zone
2:1 Duct to Inlet Diameter
Time-dependent Case : Duct Length = 4 times Duct Diameter
Steady State Case : Duct Length = 8 times Duct Diameter
x

Figure 5.1 Axisymmetric Test Case Geometry for Time-dependent and Steady State Compressible Flows

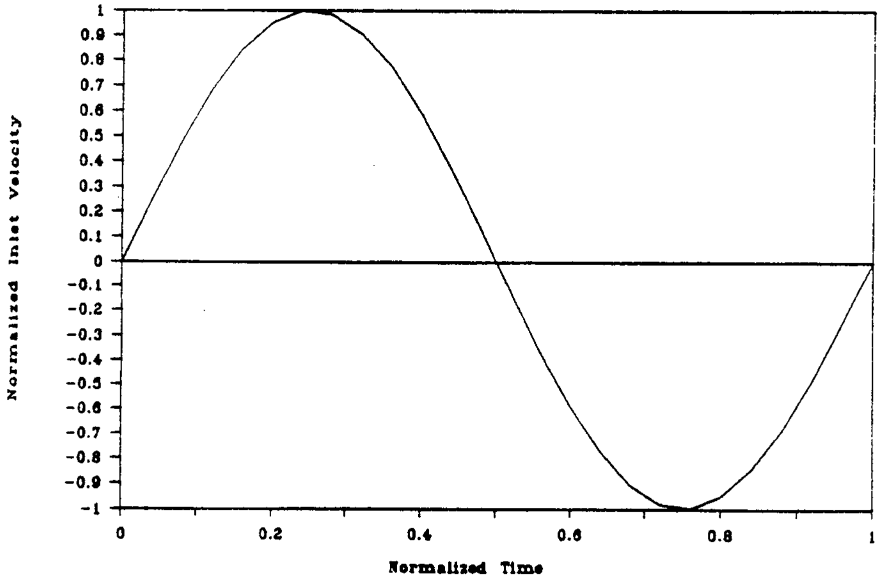


Figure 5.2 Sinusoidal Inlet Velocity Normalized by Peak Velocity V'

For the time-dependent calculations, a time step size will be defined by dividing the period of the inlet velocity cycle into a finite number of uniform increments. The number of time steps per cycle will then be a parameter used to characterize calculated results in comparisons between the performance of the two algorithms.

2. Treatment of the Boundary Conditions

Along the walls and closed end, the usual no-slip conditions hold for the momentum conservation equations. The velocity components normal to the solid boundaries are prescribed as zero due to the impermeability of the walls. For the PISO algorithm, all intermediate values of the normal component of velocity, namely, u^*, u^{**}, and u^{***} (or v^*, v^{**}, and v^{***}) are also set to zero at the wall.

When a normal component of velocity is prescribed at the boundary, it follows that for both the SIMPLE and PISO algorithms no information about the corresponding pressure gradient is needed at that boundary; thus, the appropriate "a" coefficient may be set to zero in the pressure-correction (increment) equations (4.14), (4.24), and (4.30) for the SIMPLE, PISO first corrector, and PISO second corrector calculations, respectively.

For calculation of the energy conservation equation, the wall temperatures are taken to be constant at the same value as that at inlet. This condition implies that heat transfer occurs at each of the walls.

3. Solution Procedures

For the time-dependent flow calculations, each of the algorithms generally follow their respective methodologies as outlined in Chapter III. Of utmost concern here is the attainment of accurate solutions at each time step. For time-accurate solutions, iteration at each time step is an absolute necessity for the SIMPLE algorithm, whereas the non-iterative PISO method employs only a two-stage predictor-corrector calculation once per time step. Consequently, differences in these two methodologies dictate separate requirements for the degree of accuracy to which the set of linear algebraic equations (3.2), which are the linearized representations of nonlinear fluid flow equations, must be solved.

The SIMPLE algorithm does not require highly accurate intermediate values of the field variables as iteration within each time step solves the full set of conservation equations many times over, eventually converging to sufficiently accurate solutions. Consequently, its TDMA solutions to the linear algebraic equations require only moderate accuracy, thereby permitting the specification of a small fixed number of TDMA solution iterations (sweeps). For the SIMPLE algorithm, the number of fixed TDMA sweeps is dependent upon the equation being solved: for u-velocity, v-velocity, and thermal energy equations, the TDMA is solved 3 times, while for the pressure-correction equation the number of TDMA sweeps is 5. This indicates that the TDMA is invoked a total of 14 times per iteration for the SIMPLE method.

The PISO algorithm, however, requires very accurate intermediate solutions for the field variable ϕ since the non-iterative scheme requires that continuity be sufficiently satisfied at each stage. Thus, it is imperative that each set of linear algebraic equations be adequately solved by the TDMA. The PISO algorithm ensures this by requiring that successive values of ϕ obtained after the (k^{th}) and (k^{th} + 1) TDMA sweeps possess a maximum percent difference less than a specified criterion, namely, 0.1%. This practice requires repeating TDMA sweeps until satisfactory ϕ solutions are obtained, thus no fixed number of TDMA sweeps can be specified. More efficient methods such as line successive over-relaxation (SOR) or coupled strongly implicit procedure (CSIP) of Zedan and Schneider (1985) may be incorporated to reduce the computational effort required to produce acceptable ϕ solutions.

B. Steady State Compressible Flow

1. Demonstration Problem Geometry

For the steady state compressible flow case, a similar geometry to that of the time-dependent case is used, except that now the downstream end is opened such that fluid may escape. The duct length is taken as 8 times duct diameter in order to account for the longer recirculation reattachment length of the open-end flow. The inlet velocity is now assumed not to vary with time and is taken as the same peak velocity V used before corresponding to Reynolds number of 1000 and Mach number of 0.2. Here intermediate temporal accuracy of the flow field is of no concern, and only the steady state solution is sought. For the purpose of defining a time step size, a characteristic time will be defined as $\tau = L/V$, where L is the length of the duct and V is the peak velocity. The time step size is now defined by the uniform time increment obtained by dividing the characteristic time, τ, by a finite number of time steps.

2. Treatment of Boundary Conditions

Except for those prescribed for the outlet boundary, the boundary conditions are handled identically as discussed in section V.A.2. For momentum boundary conditions at the outlet, the normal gradient of velocity is set to zero. In this study, the outlet u-velocity component is set equal to its adjacent upstream u-velocity component and the outlet v-velocity component is set equal to zero. The zero normal gradient of temperature at outlet applies for the thermal energy equation. In the PISO algorithm, it is important that the outlet density is explicitly set equal to its adjacent upstream density value, which corresponds to zero normal gradient of density at the outlet.

3. Solution Procedures

Unlike the time-dependent compressible flow case, transient calculations of the field variables are not required to be time-accurate. The SIMPLE algorithm now omits the extra iterations required to obtain highly accurate solutions at each time step, and considers each iteration equilvalent to marching forward one time step. Steady state conditions are assumed to exist when variation of the field variables is negligible over many time steps. In this study, the axial velocity components are monitored in three locations for determination of steady state conditions; these being at the recirculation zone location, the centerline zone location, and at 50% radius in the outlet plane. Steady state was assumed when the maximum standard deviation, computed over 5 time steps, of the three locations was less than 0.01.

The TDMA solution of the set of linear equations by the SIMPLE method is identical to the procedure previously discussed calculation in section V.A.3, whereas the PISO algorithm was optimized for efficiency by testing percent difference convergence criteria of 10%, 5%, 1%, and 0.5%, as well as its original 0.1%.

CHAPTER VI PRESENTATION AND DISCUSSION OF RESULTS

A. Time-dependent Compressible Flow

Calculations were first performed on a relatively coarse 20x20 grid using the iterative time-marching SIMPLE algorithm to determine the time step size, dt, at which a time-step independent solution could be assumed. Figures 6.1 and 6.2 show the predicted transient behavior of the centerline and recirculation zones, respectively, by plotting the axial velocity, normalized with respect to peak sinusoidal velocity V, versus the time coordinate, normalized by the period of the inlet velocity cycle. A time-step independent solution is seen to exist at 750 time steps per cycle. The resulting streamline contours and corresponding velocity vector plots for both monitor locations are shown in Figures 6.3 through 6.6. Figures 6.3 and 6.4 correspond to velocity conditions at the normalized cycle time of 0.35 and Figures 6.5 and 6.6 correspond to a normalized time of 0.70. Figures 6.3 through 6.6 is representative of the results obtained from both SIMPLE and PISO.

In Figures 6.7 and 6.8, comparisons are made between the velocity predictions calculated by PISO using the time step size 1/750 and the corresponding time step independent solution given by SIMPLE. The PISO algorithm is shown to produce results which closely mimic those of the time-marching SIMPLE. The small discrepancies are thought to be

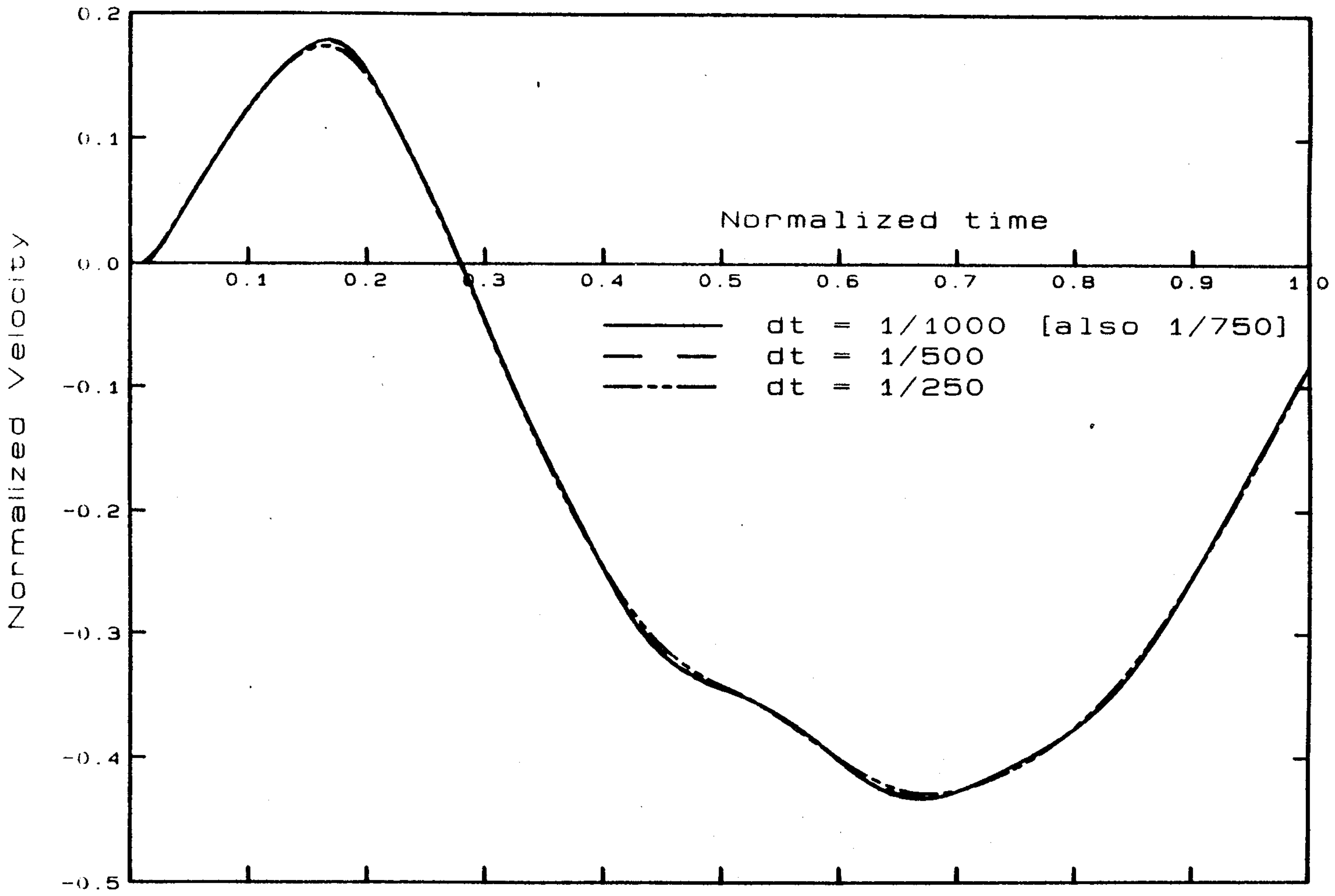


Figure 6.1 SIMPLE Velocity Predictions in Recirculation Zone for 20x20 Grid Size

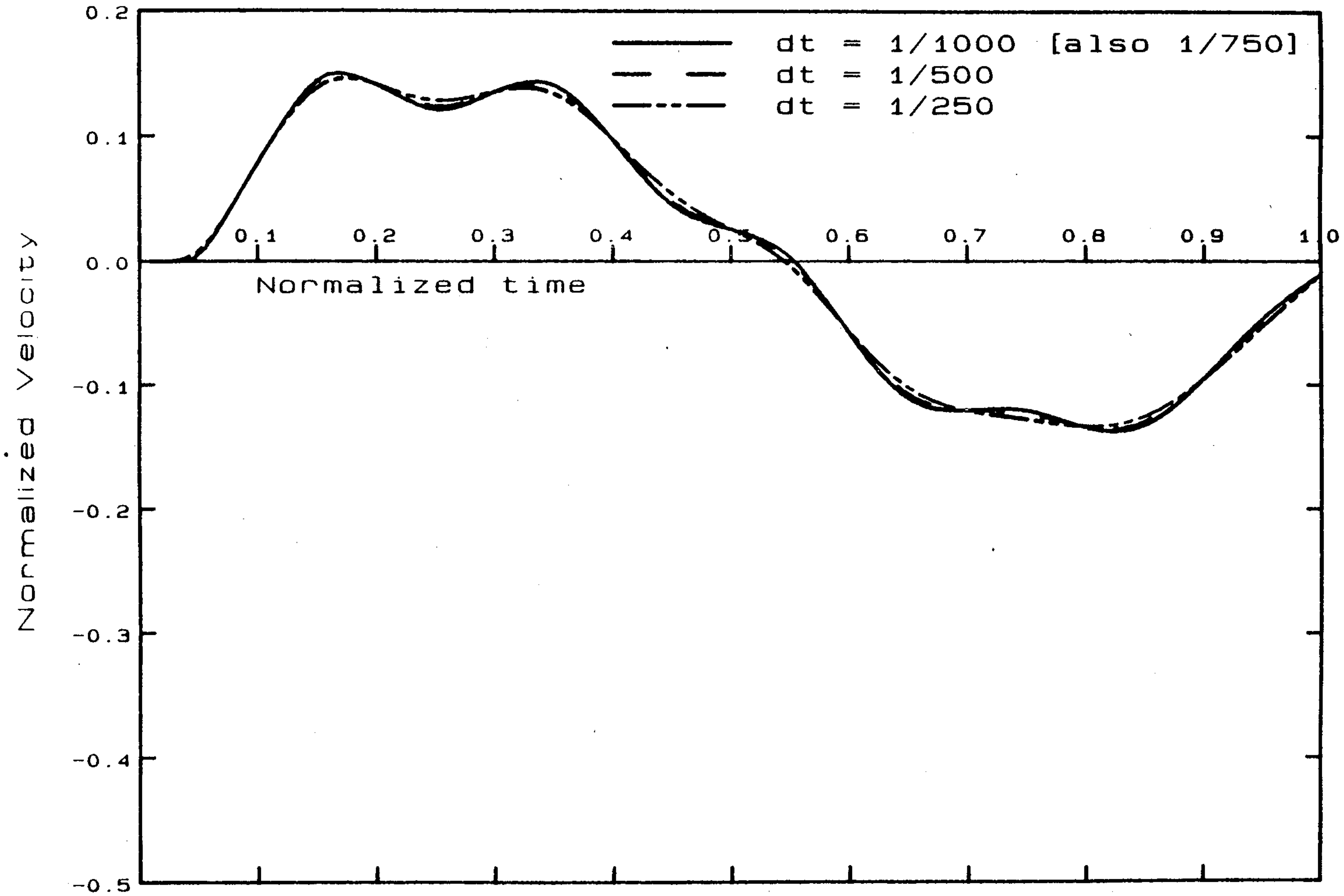


Figure 6.2 SIMPLE Velocity Predictions in Centerline Zone for 20x20 Grid Size

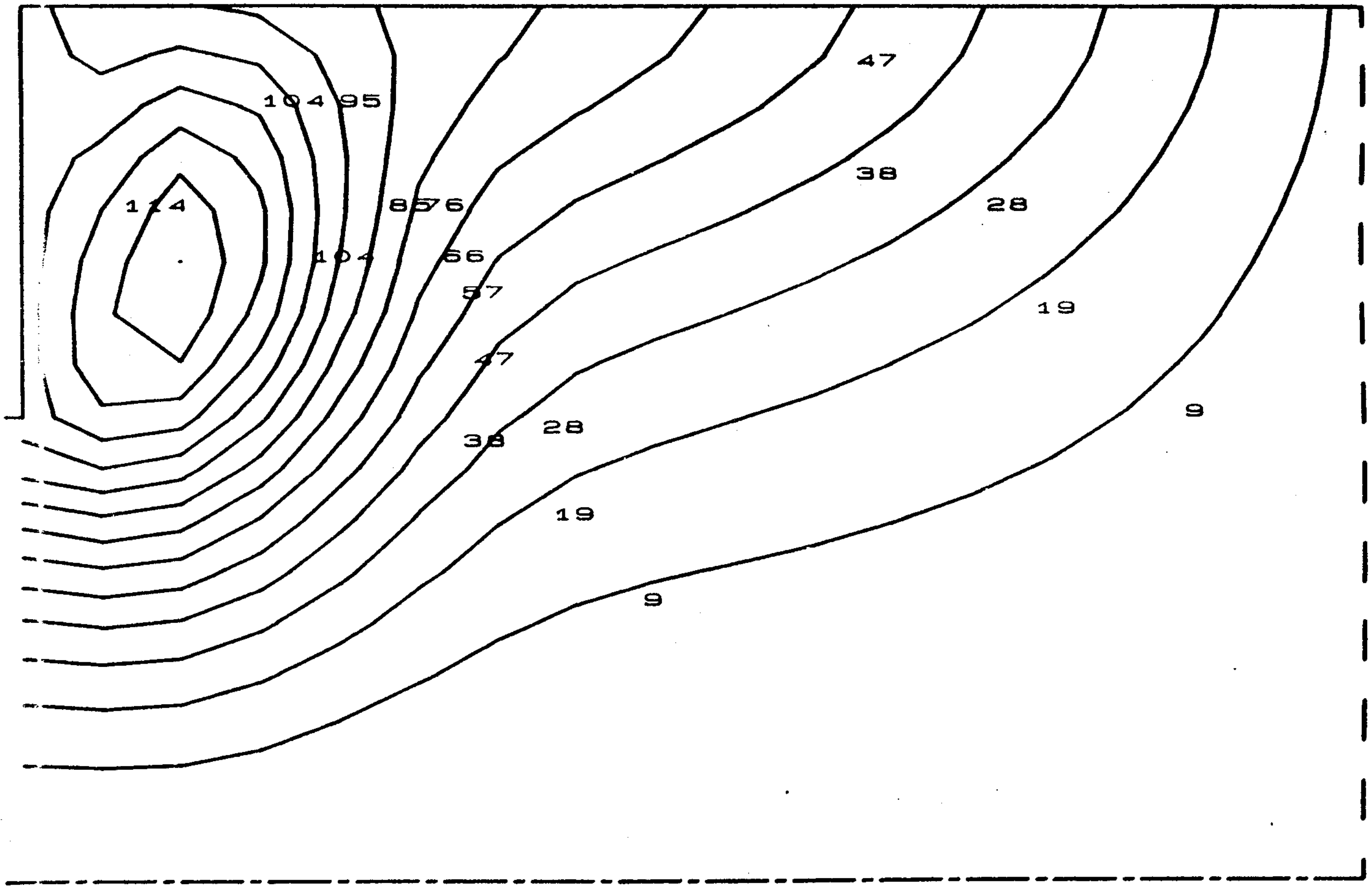


Figure 6.3 Streamline Contours for SIMPLE and PISO at 0.35 of Cycle for Case of dt=1/750 and 20x20 Grid Size

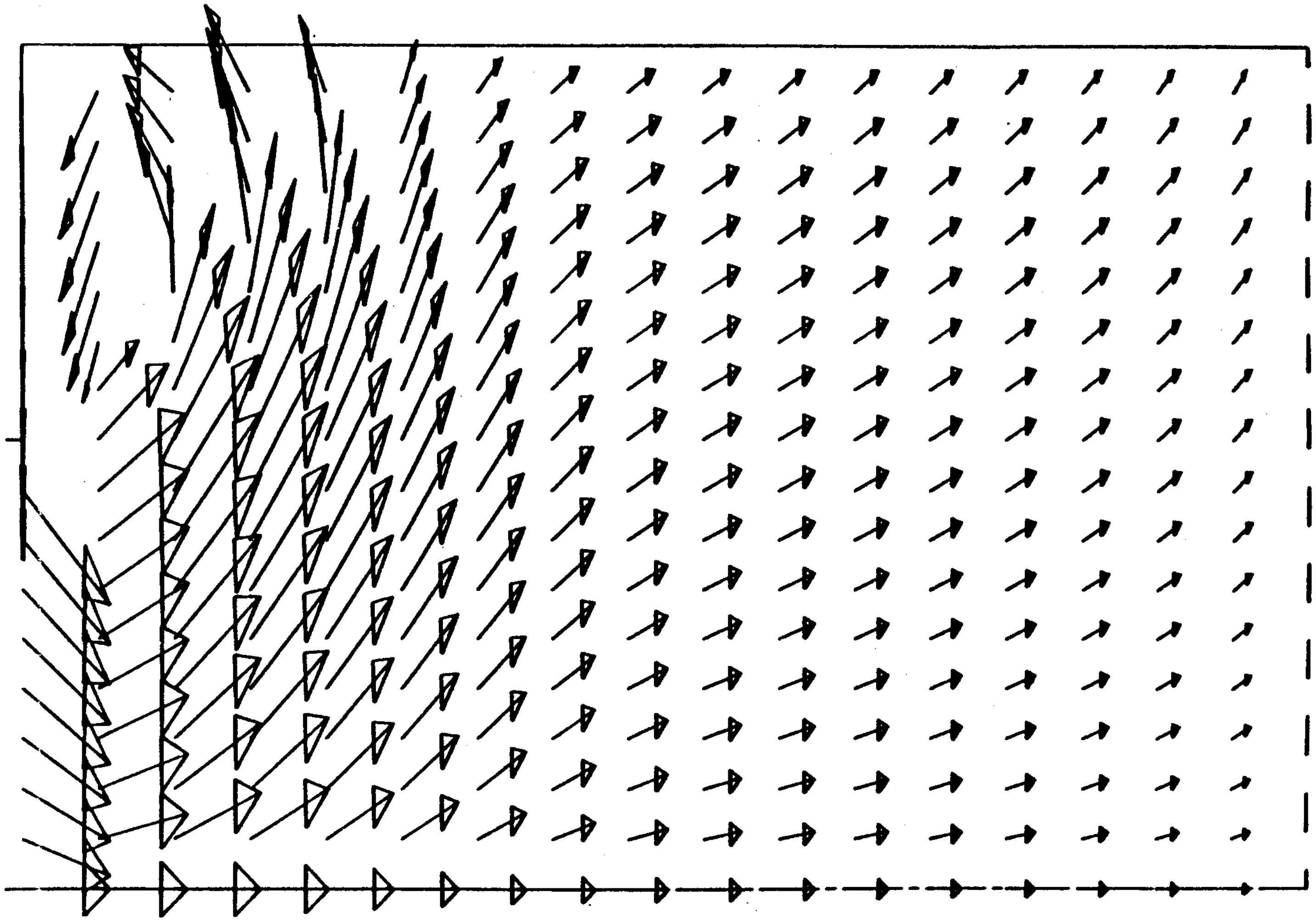


Figure 6.4 Velocity Vector Plot for SIMPLE and PISO at 0.35 of Cycle for Case of dt=1/750 and 20x20 Grid Size

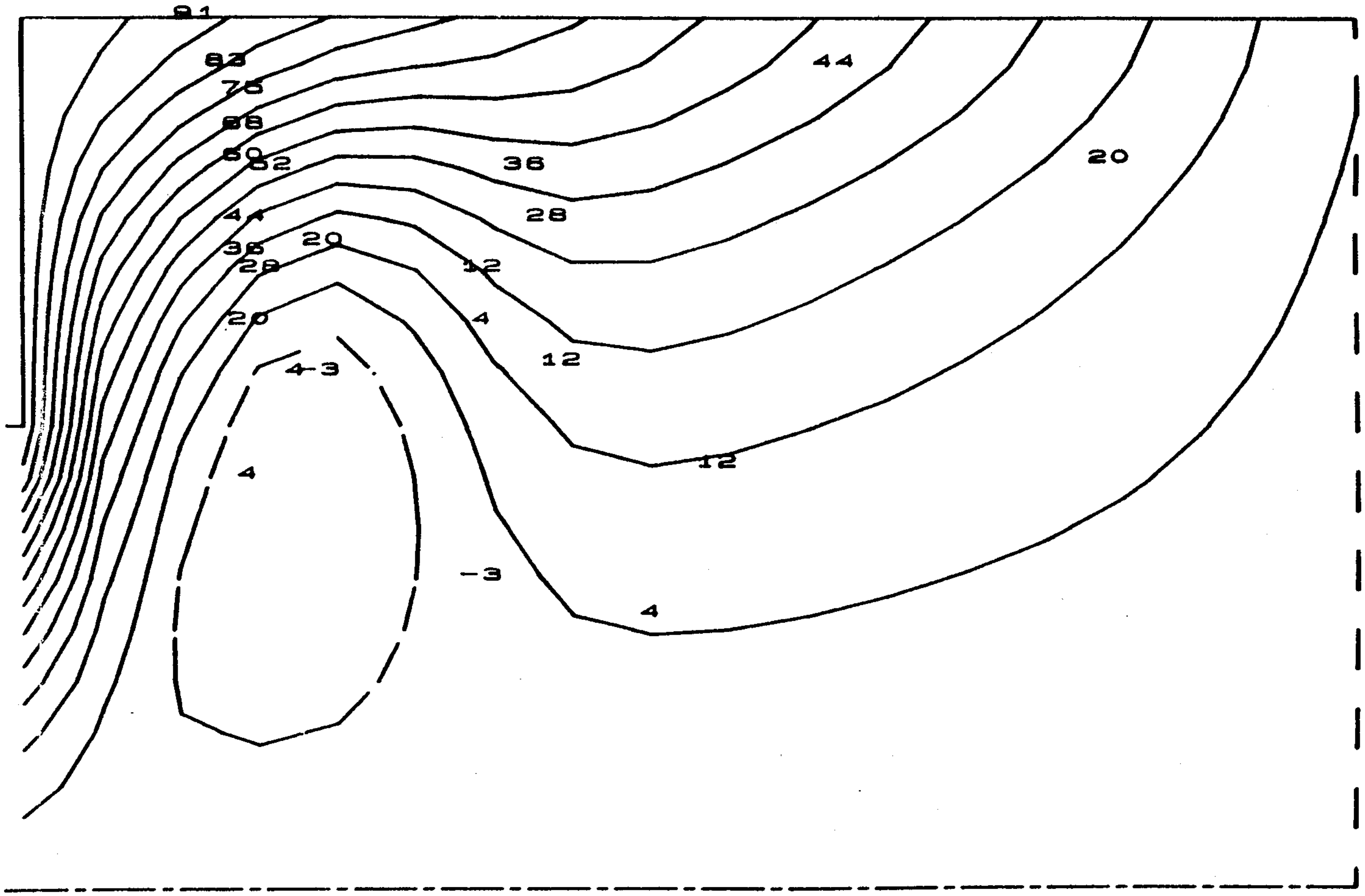


Figure 6.5 Streamline Contours for SIMPLE and PISO at 0.70 of Cycle for Case of dt=1/750 and 20x20 Grid Size

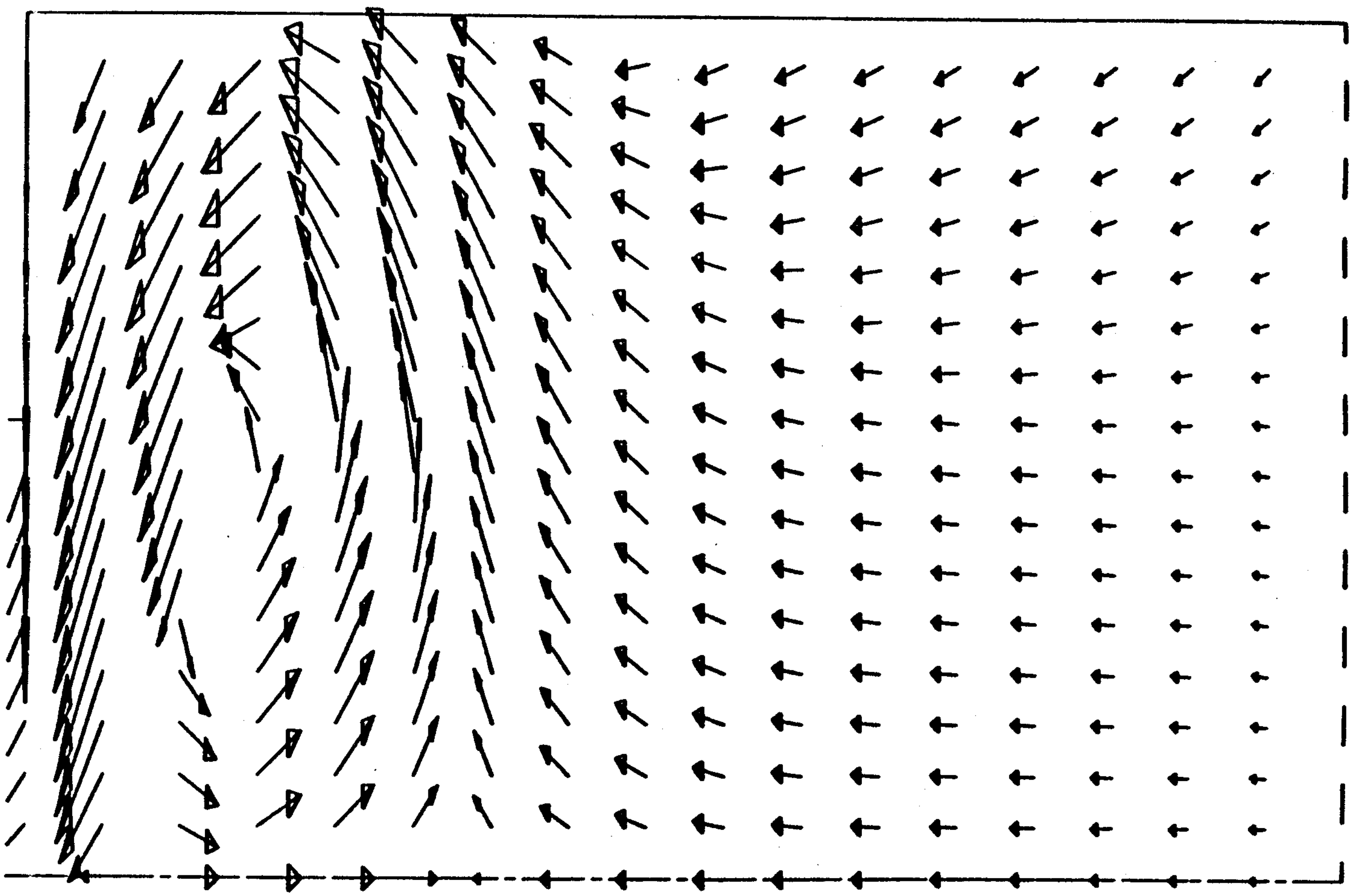


Figure 6.6 Velocity Vector Plot for SIMPLE and PISO at 0.70 of Cycle for Case of dt=1/750 and 20x20 Grid Size

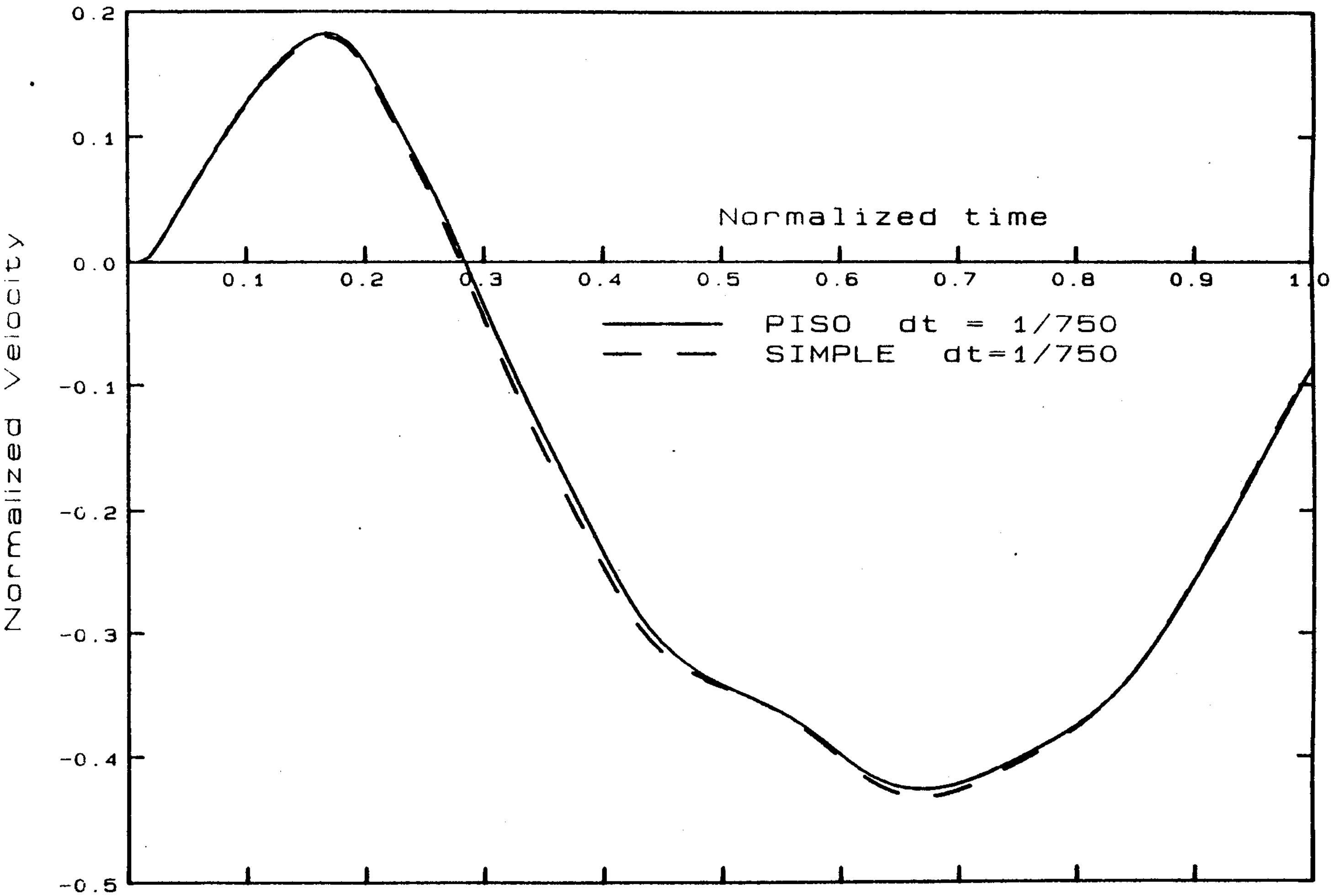


Figure 6.7 Comparison of Time Step Independent Solutions in Recirculation Zone for 20x20 Grid Size

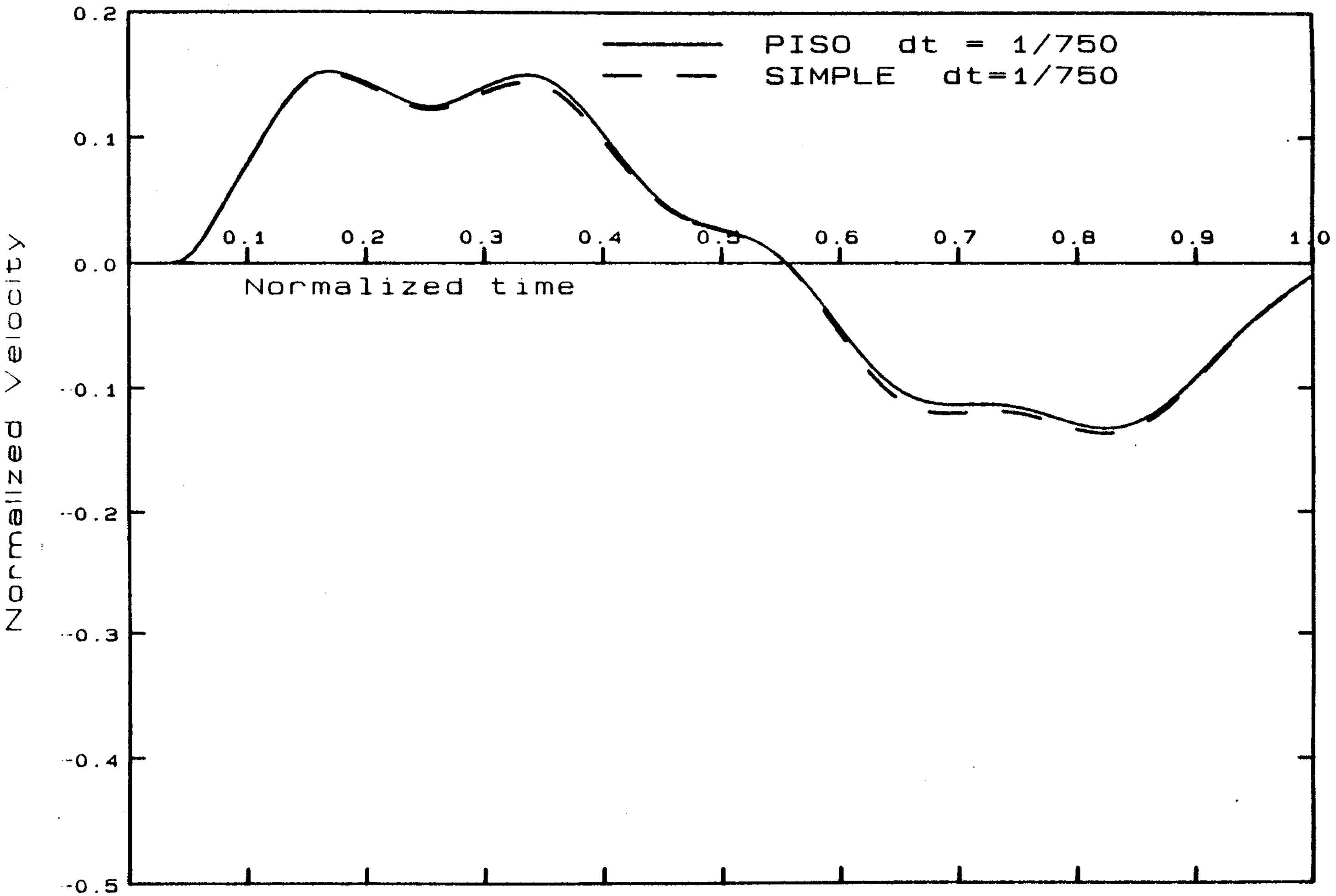


Figure 6.8 Comparison of Time Step Independent Solutions in Centerline Zone for 20x20 Grid Size

due to inaccuracies incurred from SIMPLE's practice of treating the equation of state as an auxiliary equation, rather than, as in PISO, directly incorporating it into the derivation of a pressure field equation. By segregating the equation of state from this derivation, SIMPLE's coupling between temperature and pressure appears to be incomplete, which could thereby produce inaccuracies in all resulting field variables.

The importance of correctly treating SIMPLE's temperature-pressure coupling is clearly obvious when these effects, namely, the expansivity correction terms of equation (4.12), are neglected during flow calculations. Figures 6.9 and 6.10 display the transient velocity predictions obtained when these expansivity effects are neglected (i.e. pressure-temperature coupling neglected) in equation (4.12) and (4.14). The resulting pressure field is severely underpredicted throughout the calculation domain as is indicated by Figure 6.11, which physically corresponds to the pressure field resulting from near isothermal conditions. Based on the existence of deviant SIMPLE solutions when the expansivity effects in equation (4.12) are omitted (or diminished), it is questioned whether SIMPLE's temperature-pressure coupling scheme, as implemented in this study, is as strong or accurate as that of the rigorously developed PISO algorithm.

One final note should be made regarding SIMPLE's generalized pressure-velocity coupling scheme and its

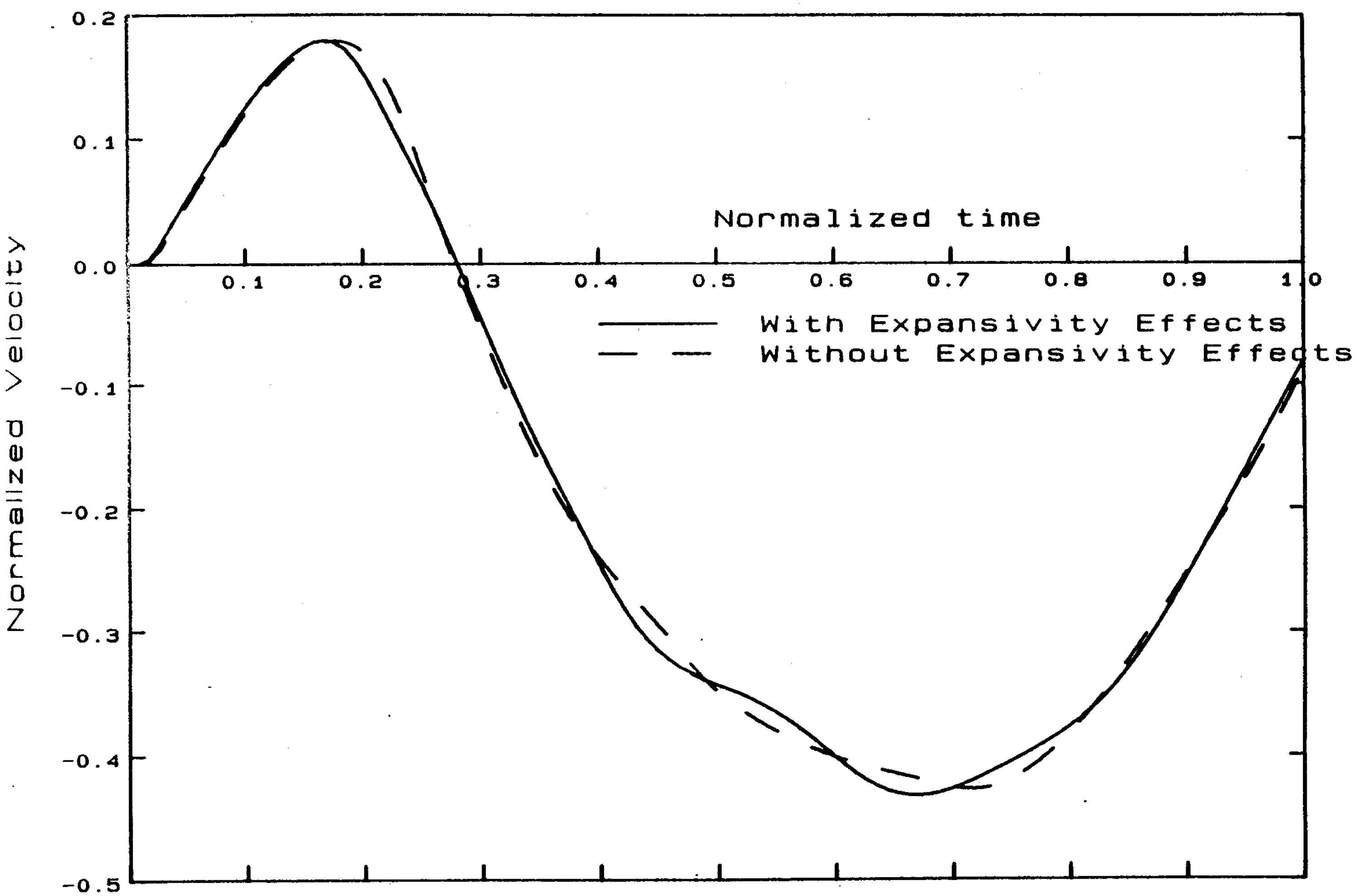


Figure 6.9 Comparison of SIMPLE Recirculation Velocity Predictions With and Without Expansivity Effects for Case of dt=1/750 and 20x20 Grid Size

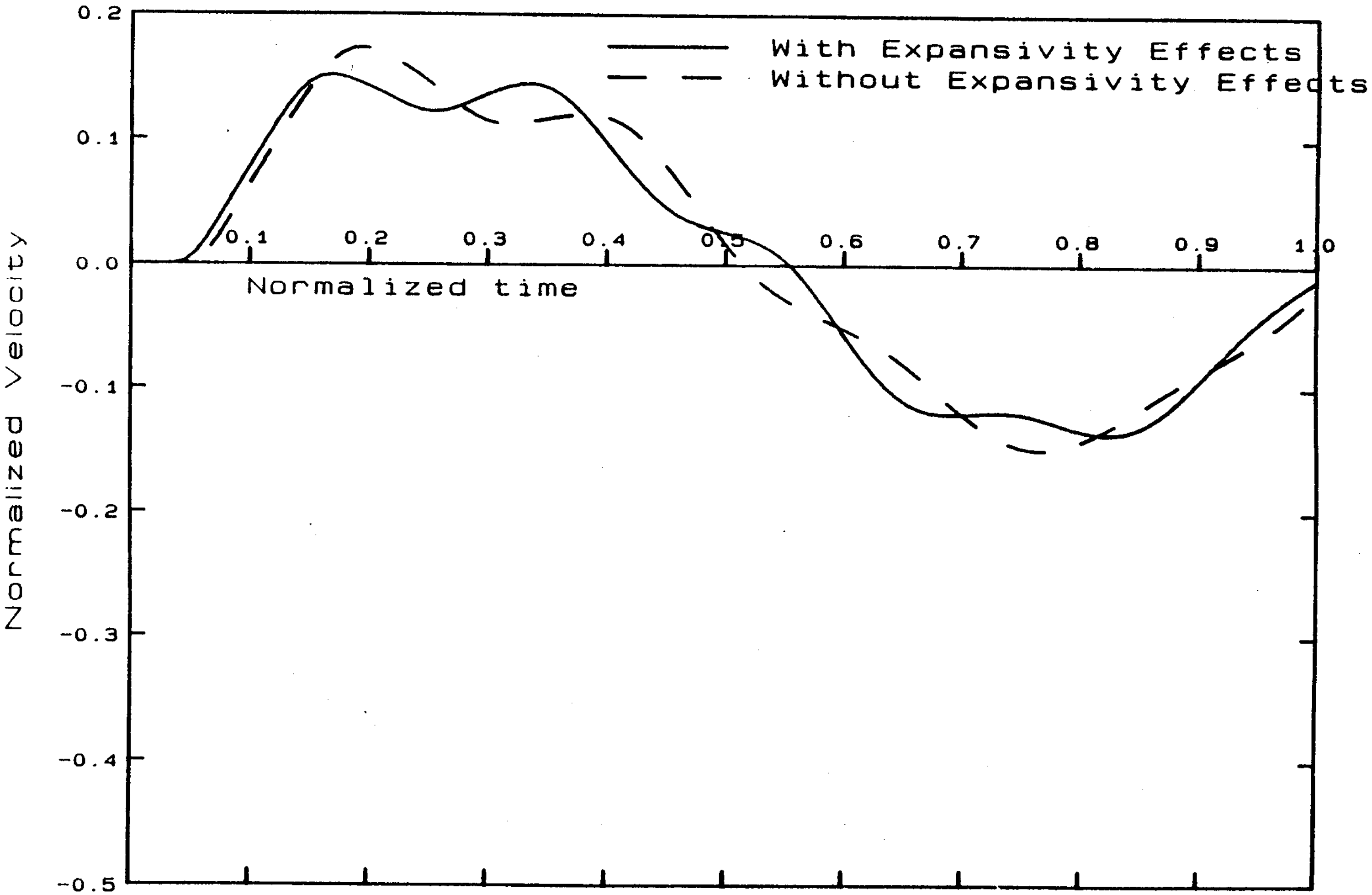


Figure 6.10 Comparison of SIMPLE Centerline Velocity Predictions With and Without Expansivity Effects for Case of dt=1/750 and 20x20 Grid Size

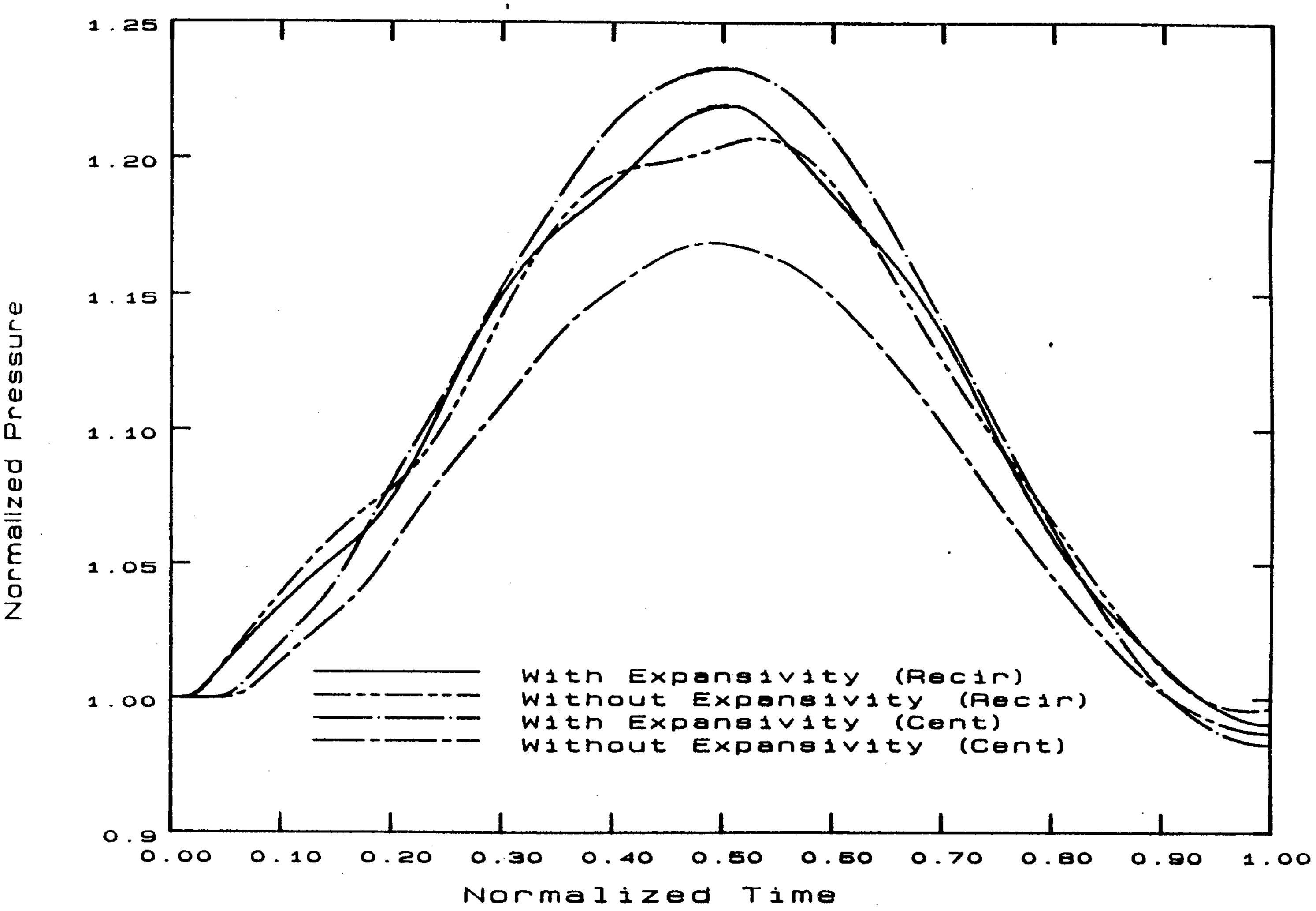


Figure 6.11 20x20 SIMPLE Normalized Pressures With and Without Expansivity for Case of dt=1/750

related expansivity correction terms. It was found necessary that the complete form of equation (4.12) be invoked for temperature-pressure coupling, whereas no obvious difficulties arose when the expansivity source term b_T in equation (4.14) was omitted. A slight computational time decrease (5%) in time-dependent computations was experienced when the b_T source term was neglected, while negligible discrepancies were observed in the resulting field variable values over the time cycle.

Figures 6.12 and 6.13 illustrates the time step independent nature and applicability of PISO for steady state analysis in that very large time steps can be taken with no stability difficulties in the solution procedure. Although the SIMPLE algorithm also indicates the ability to handle larger time steps, the non-iterative nature of PISO endows it with a favorable advantage of producing stable results at lower computing times as indicated by Table 6.1. The PISO algorithm consistantly requires 1/3 the computing effort as does SIMPLE, regardless of the time step size employed. Similar computational savings hold for results obtained using the SIMPLEC enhancement to the SIMPLE algorithm, in which negligible computational time differences are observed between SIMPLE and its variant, SIMPLEC.

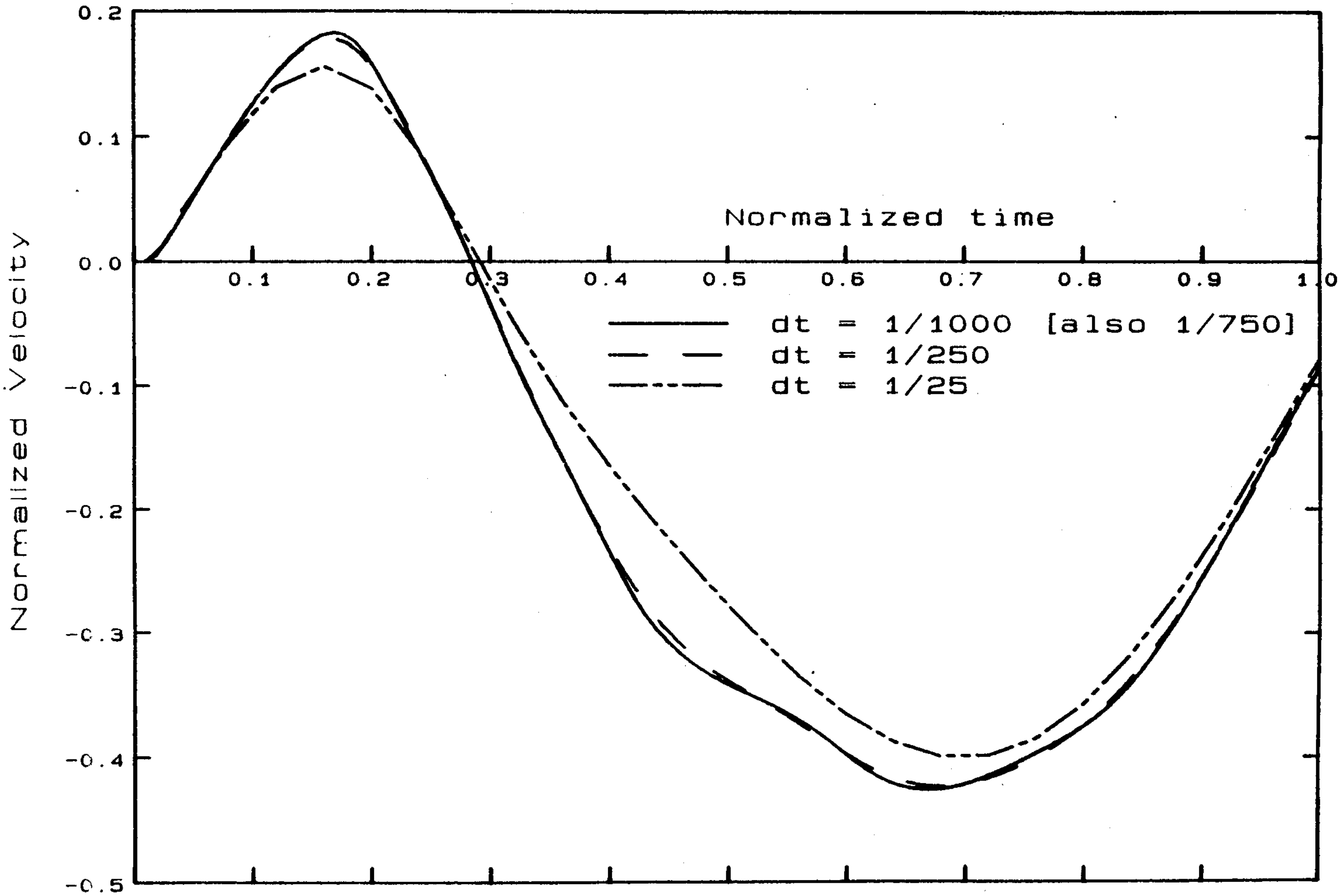


Figure 6.12 PISO Velocity Predictions in Recirculation Zone for Case of 20x20 Grid Size

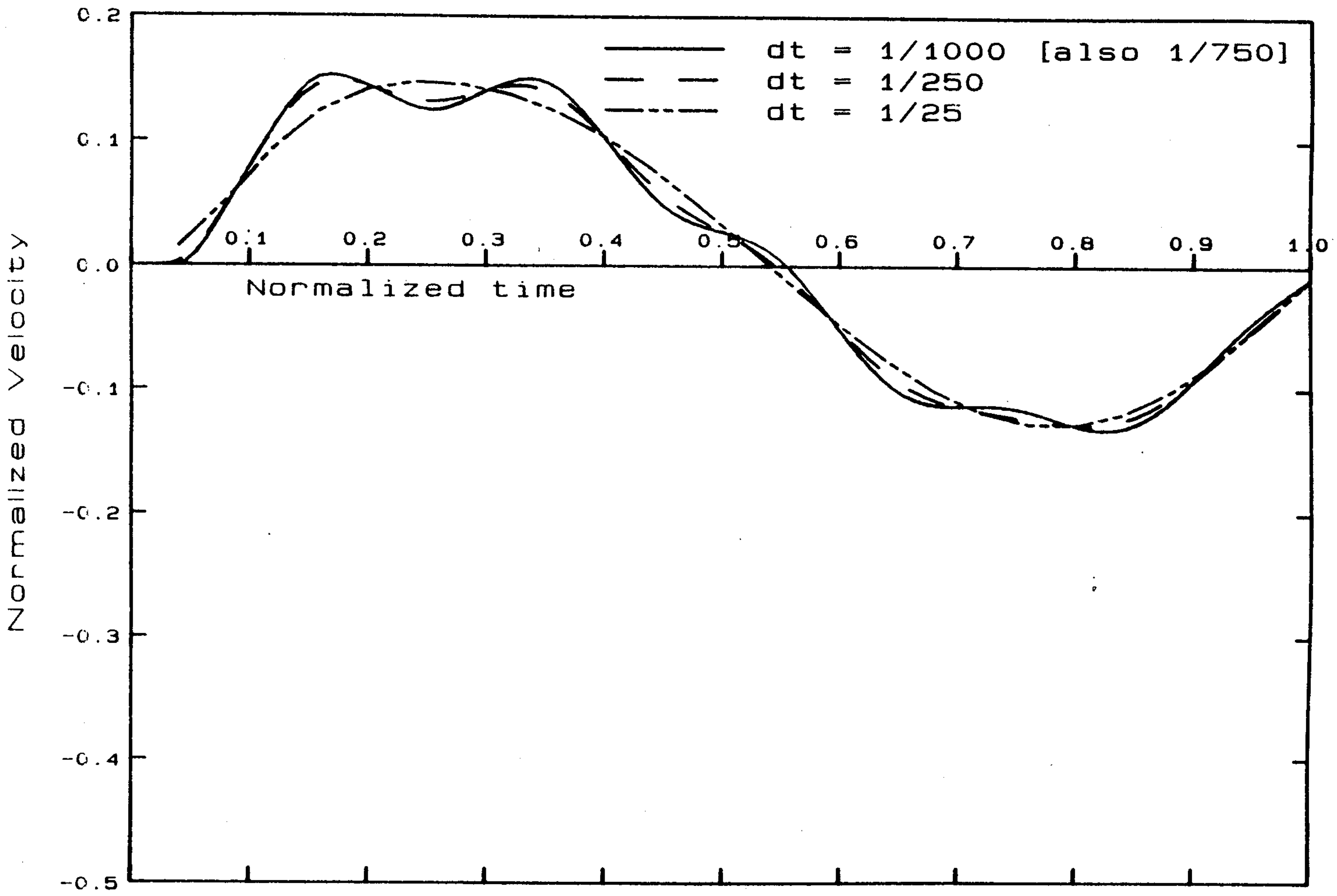


Figure 6.13 PISO Velocity Predictions in Centerline Zone for Case of 20x20 Grid Size

Table 6.1 Comparison of Computational Efficiency for Various Time Step Sizes

Solution Algorithm	Time Steps per Cycle	CPU Time (mins)	PISO/SIMPLE CPU Time Ratio
PISO	1000	60	0.328
SIMPLE	1000	183	
PISO	750	44	0.297
SIMPLE	750	148	
PISO	500	31	0.344
SIMPLE	500	90	
PISO	250	16	0.364
SIMPLE	250	44	

Next, the influence of grid size on the solutions of both the PISO and SIMPLE algorithms was determined. Figures 6.14 and 6.15 illustrate the transient velocity profiles obtained for various grid sizes using the PISO algorithm. The SIMPLE algorithm exhibits similar results. Notice that for each grid size used, significant discrepancies, namely, recirculation strength differences, occur at the recirculation zone location, whereas the centerline zone solution is relatively independent of the particular grid size employed. This observation can be attributed to the "numerical diffusion" associated with nonalignment of the coordinate grid with the flow direction.

The basic cause of numerical diffusion is treating the flow across each control volume face as locally one-dimensional [Patankar (1980)]. If the velocity vectors are

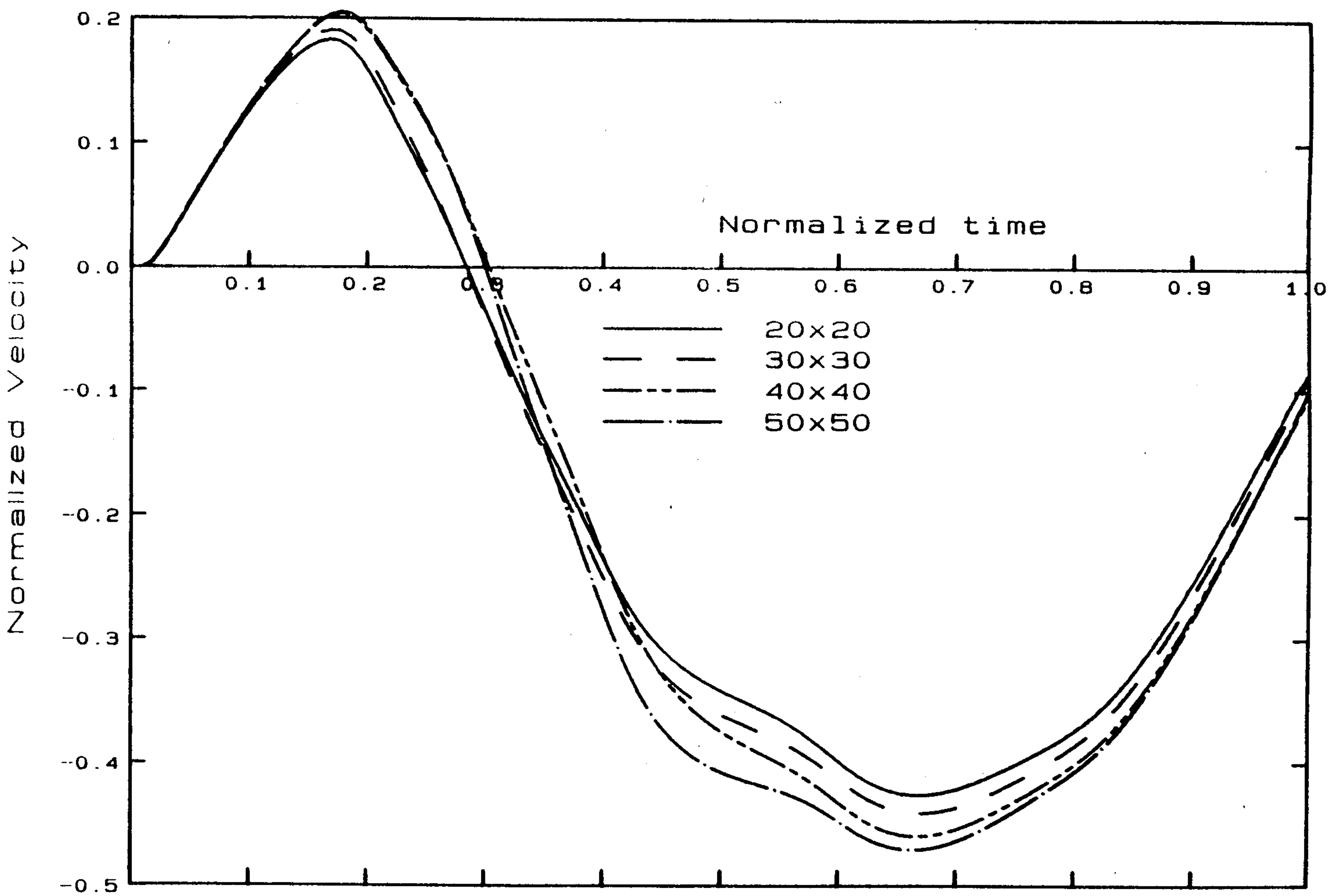


Figure 6.14 PISO Velocity Predictions in Recirculation Zone for Case of dt= 1/750 and Various Grid Sizes

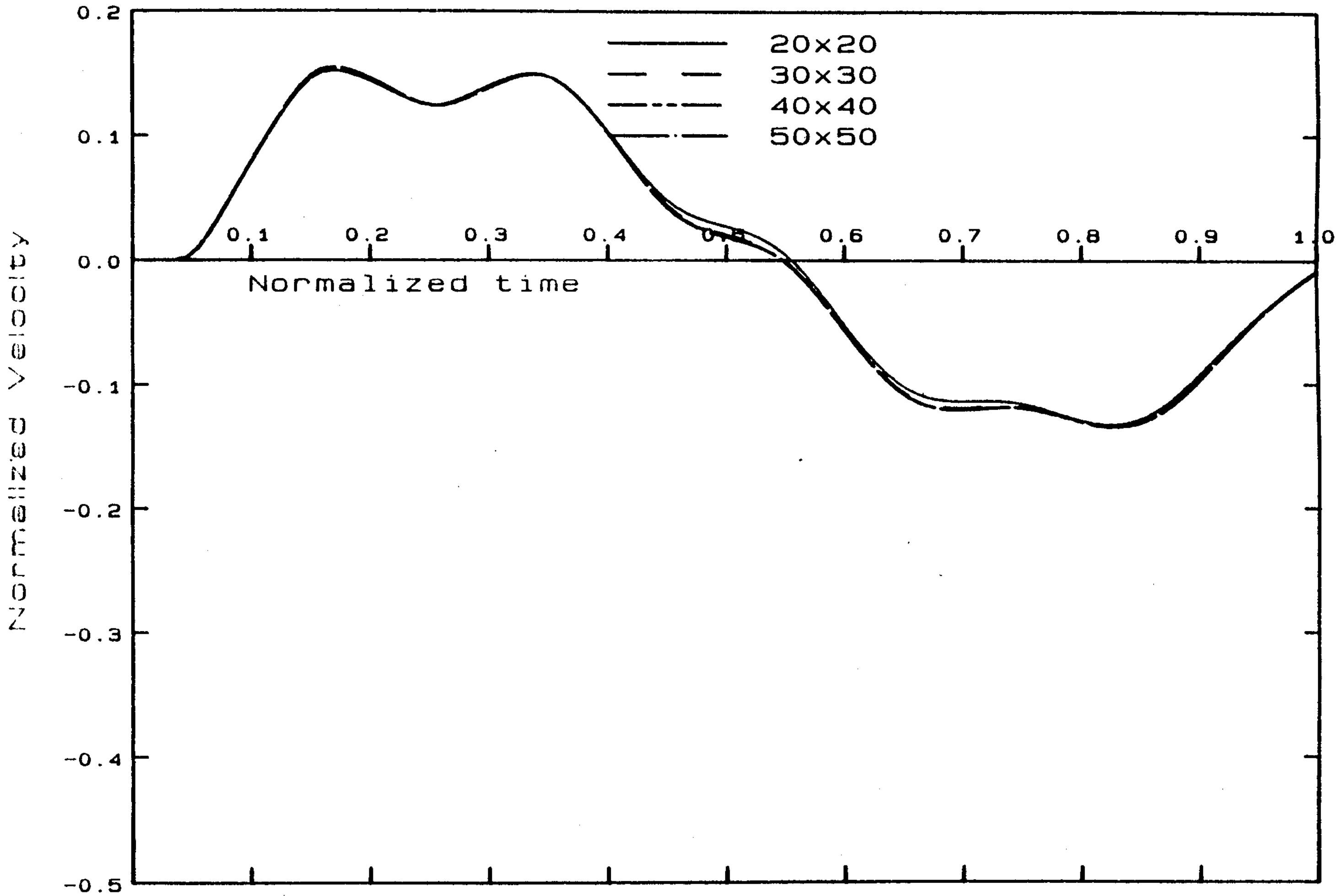


Figure 6.15 PISO Velocity Predictions in Centerline Zone for Case of dt= 1/750 and Various Grid Sizes

oblique to the grid lines, as they are in the recirculation local velocity field, then the flow into the control volume is highly skewed with respect to the control volume face and results in numerical diffusion. On the other hand, velocity vectors such as exists in the centerline zone are fairly aligned with the coordinate grid, and the magnitude of numerical diffusion is greatly reduced, thus giving almost grid independent solutions.

By using sufficiently fine grids, the numerical diffusion may be reduced to tolerable levels, but this practice proves to be impractical due to increased computational costs. In Figure 6.14 the strength of the recirculation region is seen to increase as the grid is refined, which is a result of decreased numerical diffusion leading from the finer grid sizes.

Recent works [Benodekar et al. (1985)] have found the hybrid method and its "lower order of accuracy" lacking when attempting to predict gross features of the recirculation zone, namely, underpredicting the recirculation region dimensions. Schemes which involve more neighbors in the discretization equation, such as those mentioned in section III.B of Raithby (1976) and Leonard (1979), give less numerical diffusion, but are significantly more complicated than the hybrid scheme.

Due to extraordinarily long computing times and increased storage requirements necessary for fine grids, no grid-independent solution was obtained for either algorithm.

Solutions from each algorithm, though, are compared for a case of 40x40 grid size and 750 time steps per cycle in Figures 6.16 and 6.17, which shows very close qualitative resemblances to the 20x20 grid results of Figure 6.7 and 6.8, respectively. The main difference between the 40x40 and 20x20 sets of results is an increased recirculation zone strength for the 40x40 case resulting from the decreased numerical diffusion of the more refined grid. It is interesting to note that PISO has maintained, as before, an approximate 60% time savings in computing effort for each grid size employed as shown in Table 6.2.

Table 6.2 Comparison of Computational Efficiency for Various Grid Sizes

Solution Algorithm	Grid Size	CPU Time (mins)	PISO/SIMPLE CPU Time Ratio
PISO	20x20	44	0.297
SIMPLE	20x20	148	
PISO	30x30	106	0.348
SIMPLE	30x30	305	
PISO	40x40	204	0.348
SIMPLE	40x40	586	

B. Steady State Compressible Flow

The first step in computing a steady state compressible flow case was to run a 20x20 grid size SIMPLE algorithm for various time steps per characteristic time period, namely, 25,50, and 75, in order to study the general flow

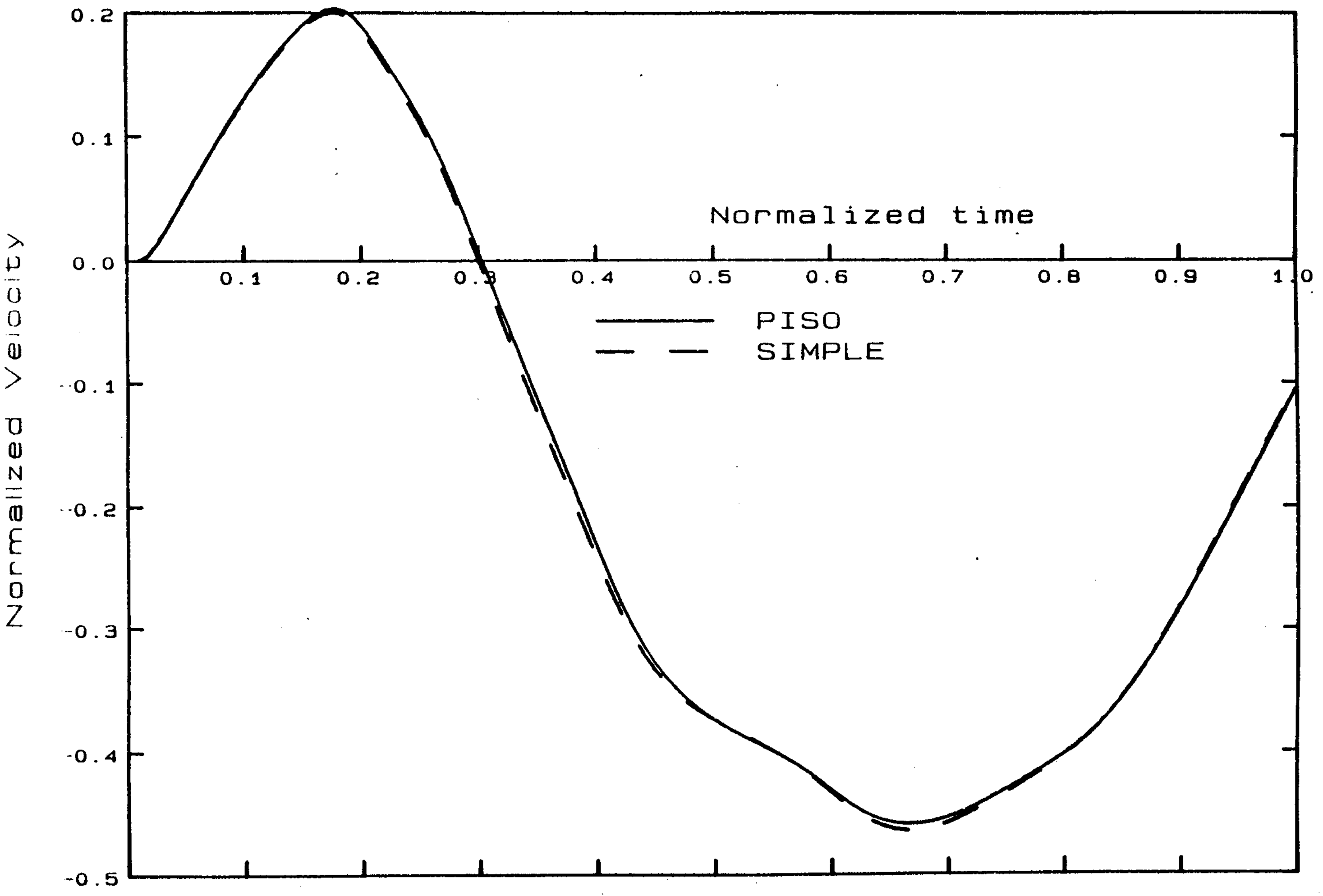


Figure 6.16 Comparison of Recirculation Zone Velocity Predictions for Case of dt=1/750 and 40x40 Grid Size

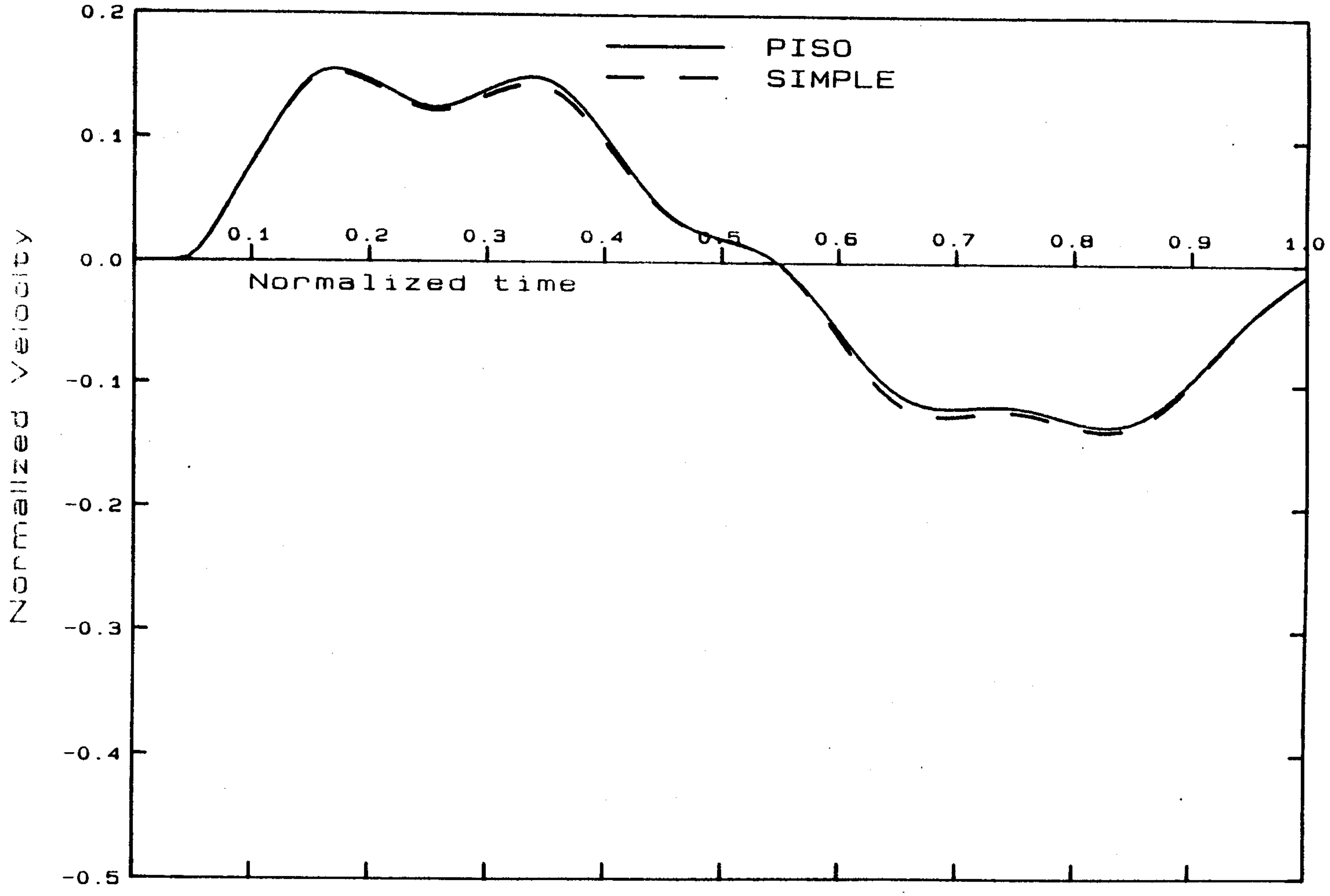


Figure 6.17 Comparison of Centerline Zone Velocity Predictions for Case of dt=1/750 and 40x40 Grid Size

field characteristics. The streamline contour plot corresponding to 75 time steps per cycle is shown in Figure 6.18 and is also representative of similar results obtained from SIMPLEC and PISO. The recirculation zone reattachment length is shown to be approximately 0.6 of the duct length, which is approximately 20 step heights. A literature search yielded no information pertaining to laminar, compressible axisymmetric pipe-expansion flow, however, the work of Milos and Acrivos (1987) was found, which have performed numerical simulations of laminar, incompressible, two-dimensional flow past a sudden expansion at large Reynolds number. From Milos et al. (1987) work, the corresponding incompressible flow situation as compared to this study's steady state compressible case, i.e. same Reynolds number and diameter ratio, displayed reattachment lengths of between 18 and 25 step heights. This indicates that the compressible flow characteristics as computed by the SIMPLE and PISO algorithm are in general agreement with similar previous work dealing with incompressible flow, even though some unknown discrepancies may exist due to the differing nature of the flows.

In the case of steady state calculations, when temporal accuracy is of no consequence, the convergence tolerance on the TDMA solution of the set of linear algebraic equations may be relaxed at each time step. With this in mind, PISO reduced the stringency of its convergence criterion, whereas, the SIMPLE method continued using the same TDMA procedure as before, i.e. 14 total TDMA sweeps per

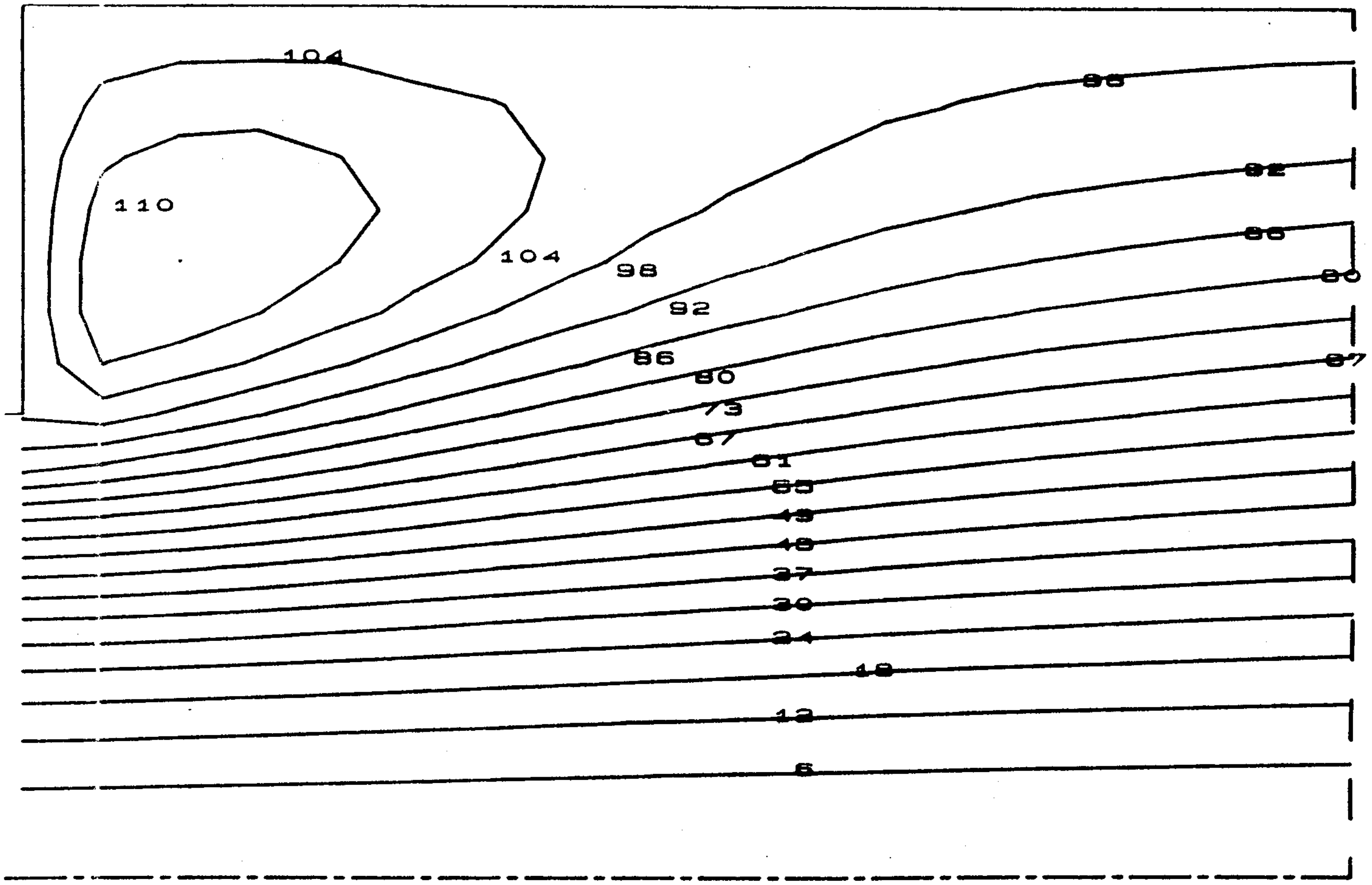


Figure 6.18 Streamline Contour Plot for 20x20 SIMPLE, SIMPLEC, and PISO at Steady State Conditions and Time Step Size of 1/75

time step. Attempts of specifying a fixed number of TDMA weeps for the PISO method yielded totally unstable solutions; therefore, the variable TDMA sweep approach was retained, and the effects of varying the percent difference convergence criterion were studied. Figure 6.19 illustrates that PISO, using 10 time steps per cycle and 20x20 grid size, was computationally most efficient at a convergence criterion of 1% for the maximum percent difference existing between solutions arising from successive TDMA sweeps.

Comparisons were then made between SIMPLE, PISO (employing a convergence criterion of 1%), and the SIMPLEC variant of SIMPLE for various time step sizes. Figure 6.20 shows that for a range of smaller time steps sizes, i.e. number of time steps per time greater than 25 and less than 75, both the SIMPLE and SIMPLEC methods exhibit favorable advantages over PISO in terms of computational effort. For the larger time steps, however, both SIMPLE algorithms diverged, whereas, the PISO algorithm remained stable and also gained a slight advantage in computational efficiency. Figure 6.20 is somewhat surprising when compared to results obtained from similar studies performed on incompressible flows by Issa et al. (1986). For incompressible steady state flows, Issa et al. (1986) found that PISO exhibited consistant computing time advantages over a steady state version of SIMPLE, and also maintained remarkable stability and robustness over a wide range of time step sizes. The PISO method, however, displays an entirely different nature

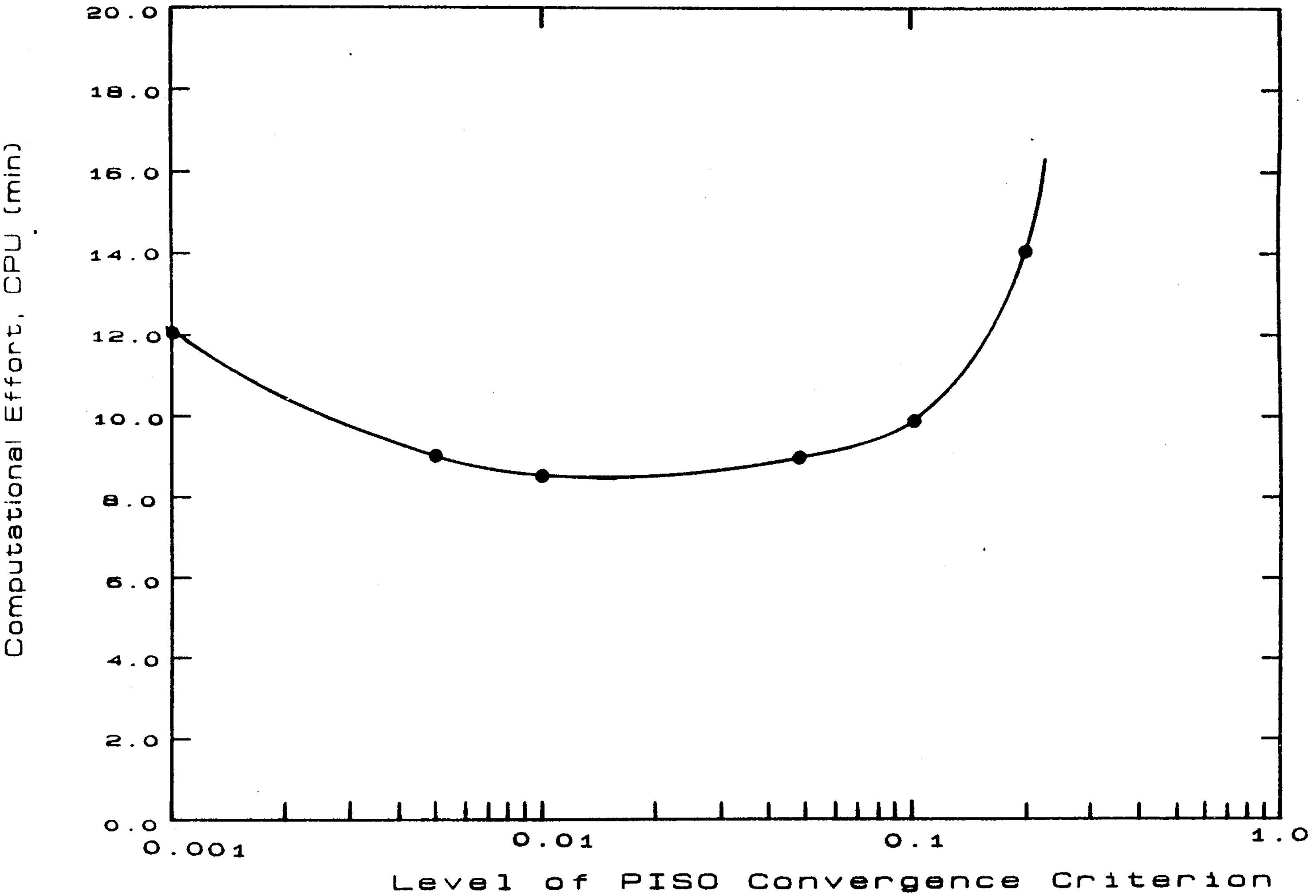


Figure 6.19 Computational Requirements versus PISO TDMA Convergence Criterion for 20x20 Grid Size and dt=1/10

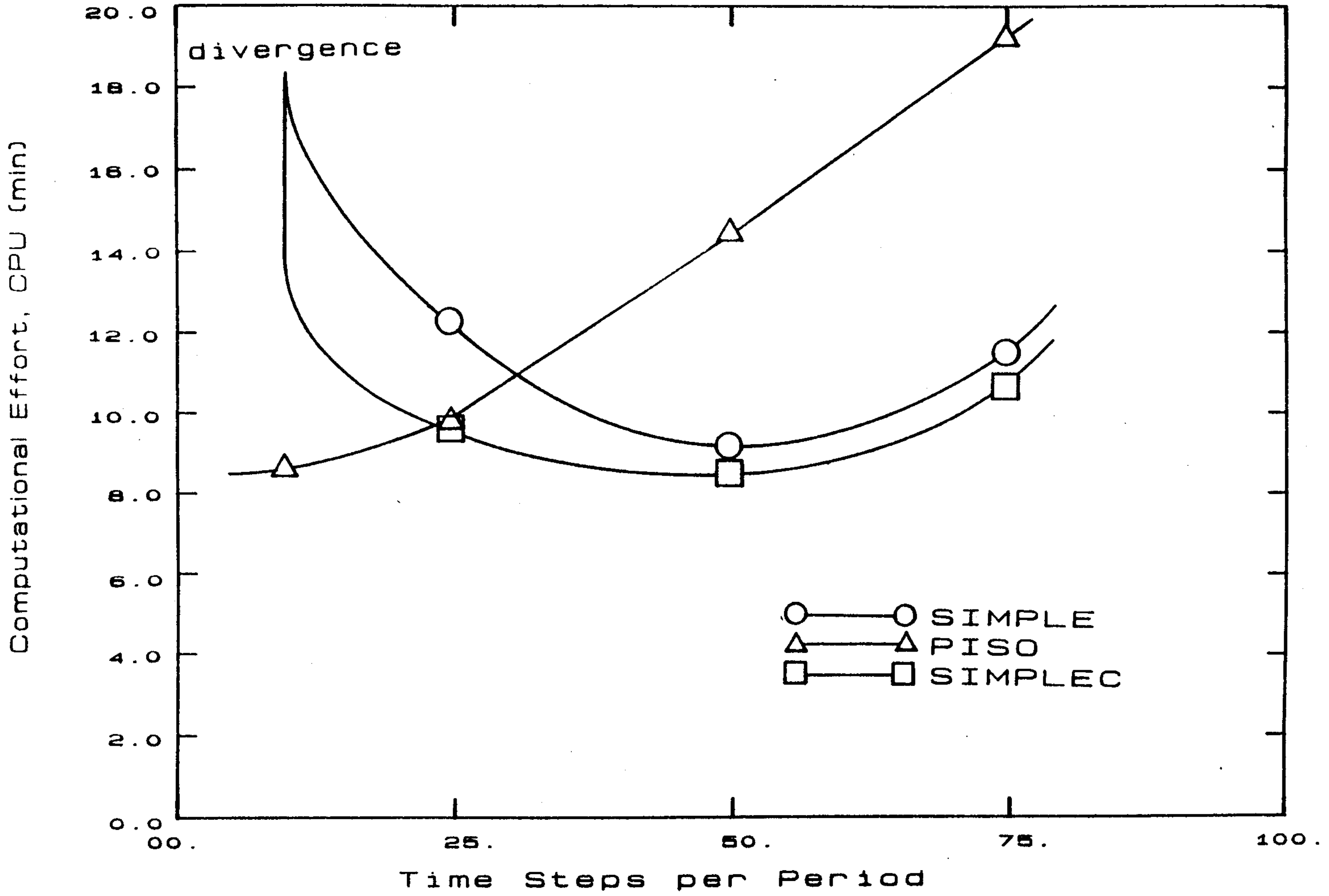


Figure 6.20 Computational Requirements versus Time Step Size for SIMPLE, SIMPLEC, and PISO for 20x20 Grid Size

in steady state compressible flow analysis, in that computational time advantages, in general, belong to SIMPLE and no robust nature is exhibited.

In an attempt to explain this sudden reversal of PISO's superiority over SIMPLE when applied to steady state compressible flow calculations, two observations can be made; first, PISO requires more computational time per time step (iteration) than SIMPLE, and secondly, SIMPLE exhibits different performance behavior for steady state incompressible and compressible flow calculations. If we examine the total number of time steps required for each algorithm to obtain steady state results versus the time step size, as shown in Figure 6.21, the statement that PISO requires more computational time per time step seems contradictory. However, Figure 6.22 indicates that PISO's total number of TDMA sweeps per time step, which is taken as a relative measure of computational effort per time step, is consistently higher than the fixed 14 total TDMA sweeps per time step of SIMPLE and SIMPLEC. Therefore, although PISO may actually require fewer, or an equal, number of total time steps to obtain steady state conditions than SIMPLE, the actual computational effort due to strict TDMA solution requirements is significantly higher at each time step.

Addressing the second point, it is essential to understand that the iterative SIMPLE method, when applied to incompressible flows whether as a time-marching or steady state version, requires substantial under-relaxation of the

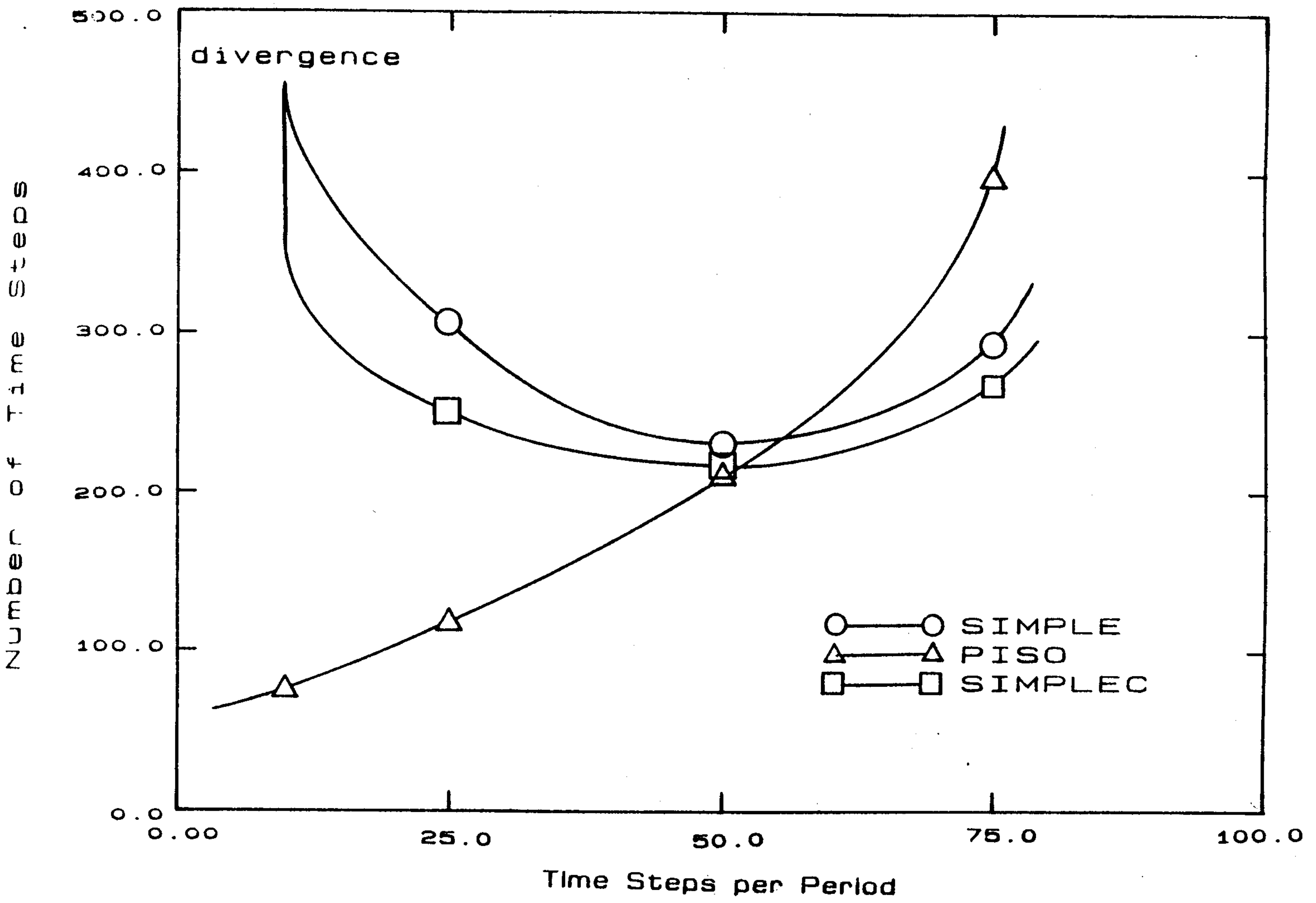


Figure 6.21 Total Number of Time Steps Required to Obtain Steady State Conditions for 20x20 Grid Size Calculations

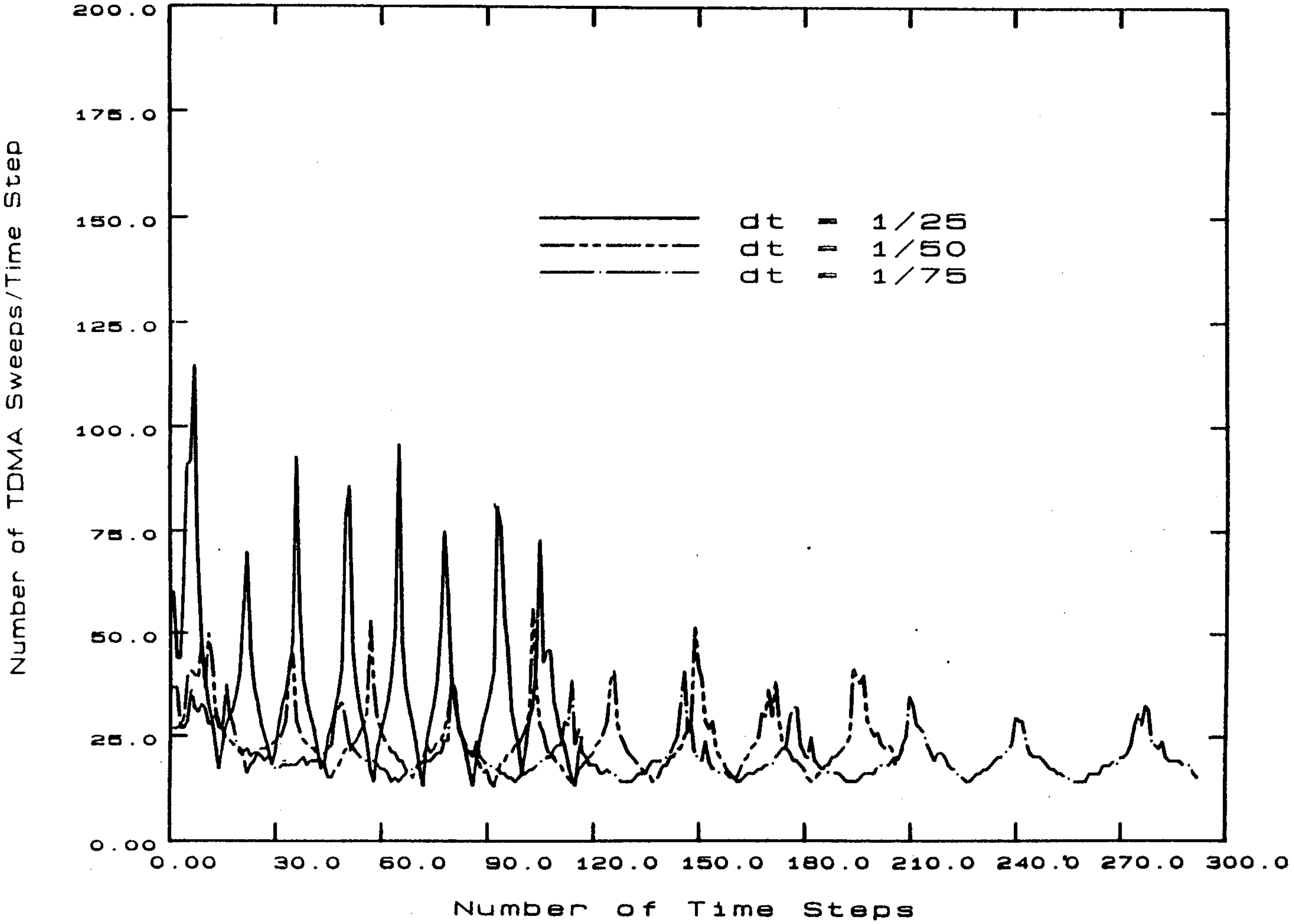


Figure 6.22 Number of PISO TDMA Sweeps per Time Step for Case of 20x20 Grid Size and Convergence Criterion of 1%

momentum equations in order to obtain convergent solutions as illustrated by Issa et al. (1986). But when applied to steady state compressible flows, all calculations performed in our study required no under-relaxation for both the SIMPLE and SIMPLEC methods and only experienced stability difficulties for which under-relaxation was ineffective. This unique characteristic is apparently due to the additional density variations of compressible flows and is not found in incompressible flows. Thus, in incompressible calculations, the total number of time steps required to obtain steady state conditions is increased due to the under-relaxation of the momentum equations as compared to its compressible flow counterpart. This increased number of required time steps is then more computationally detrimental to SIMPLE than is PISO's own disadvantage of higher computing effort per time step, and thereby establishes PISO as the most efficient algorithm for incompressible flows.

In developing a generalized flow solver, the time-dependent flow case indicated the necessity of including expansivity terms to account for temperature-pressure coupling for flows in which these effects were important. For the steady state compressible open-end pipe expansion geometry, the temperature-pressure coupling is not as strong as in a compression/expansion type process, as Figure 6.23 illustrates for temporal axial velocity variations leading to steady state conditions. For this flow case, identical steady state conditions for all field variables were

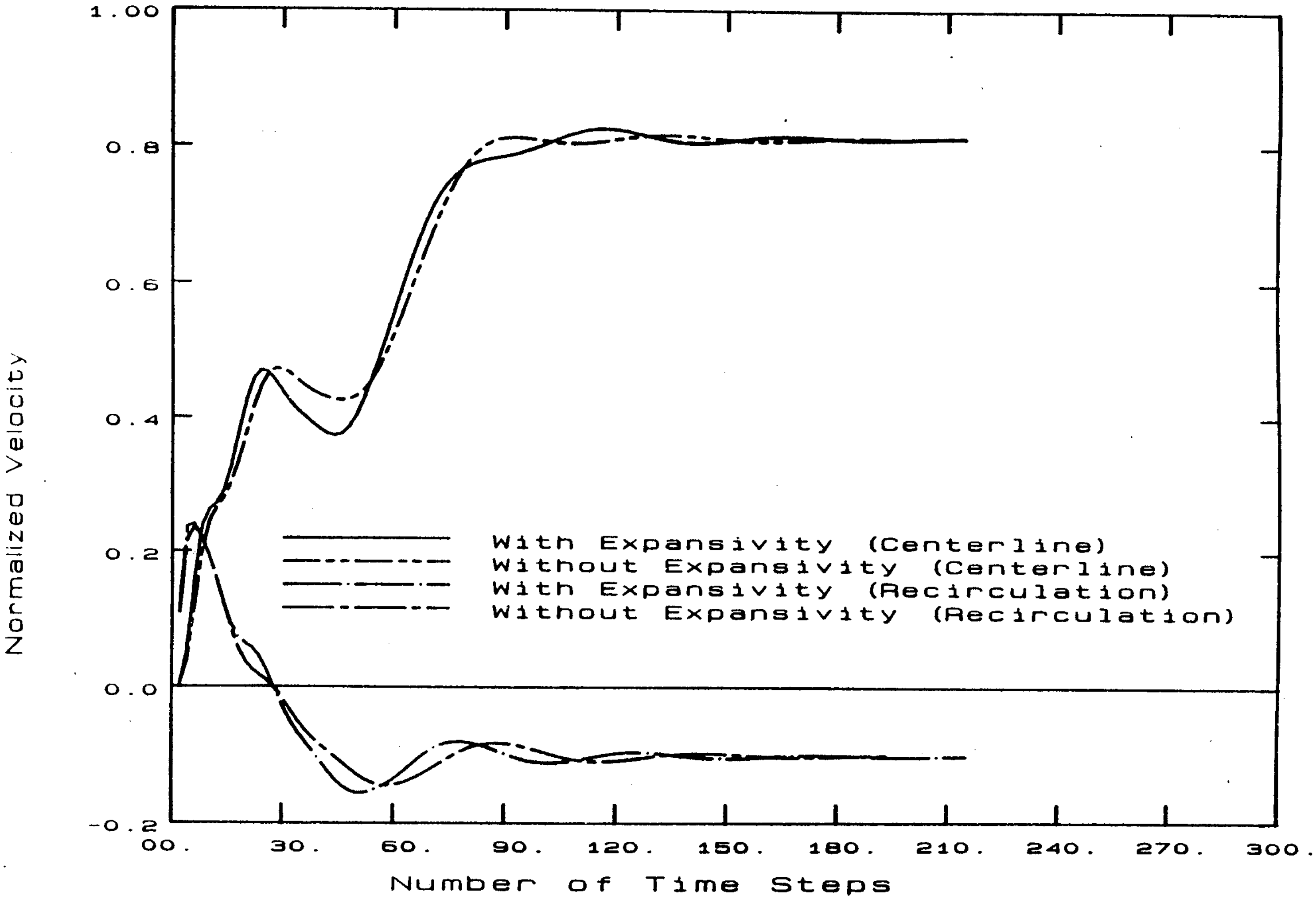


Figure 6.23 SIMPLE Temporal Velocity Variations With and Without Expansivity for Time Step Size of 1/50

achieved whether or not the expansivity terms of (4.12) or (4.14) were included in the calculations.

Figure 6.24 illustrates the streamline contour plot of flow conditions existing for calculations ran using a grid size of 40x40 and is representative of both the SIMPLE and PISO algorithm results. A time step size corresponding to 100 time steps per characteristic time period was employed due to the tendancy of SIMPLE to diverge at larger time step sizes when the 40x40 grid was used. The gross recirculation zone dimensions agree very well with those obtained using a 20x20 grid. Of significance, however, is the fact that the PISO algorithm required 3 times the computational effort as compared to the SIMPLE method to obtain steady state conditions.

One final topic of interest evolves from analysis of the 40x40 grid size case, that concerning the normalized steady state pressure plots. Figure 6.25 compares the development of the steady state normalized pressure existing at the centerline zone location for SIMPLE and PISO using 40x40 grids. It is observed that both methods yield a final normalized centerline pressure of approximately 1.01 which may be taken as an accurate reference value due to the fine grid size and small time step size employed. Examining similar plots for the 20x20 SIMPLE, SIMPLEC, and PISO algorithms in Figures 6.26, 6.27, and 6.28, respectively, it is observed that for each algorithm the final normalized centerline pressures deviate from 1.01 when the number of

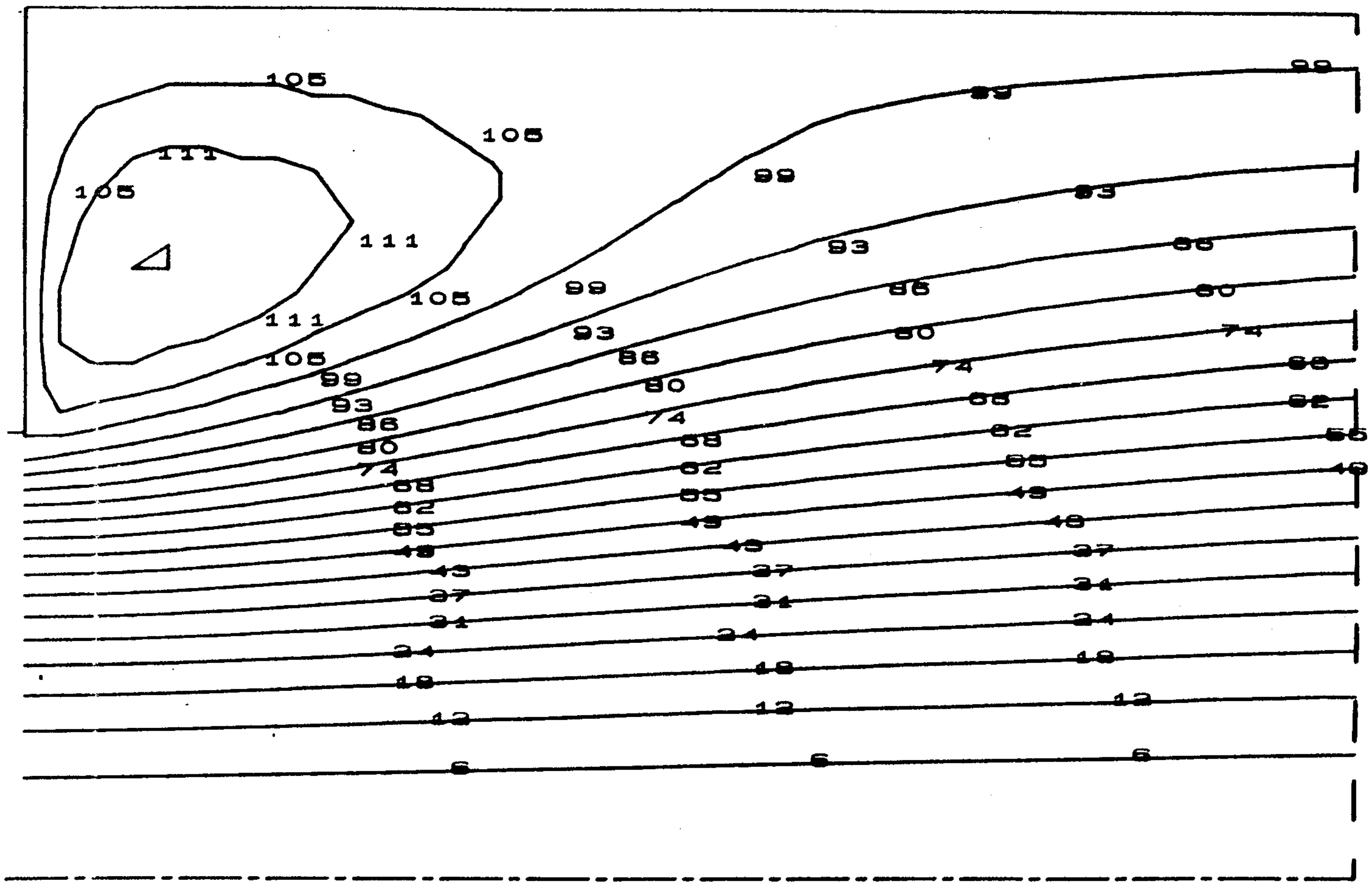


Figure 6.24 Streamline Contour Plot for 40x40 SIMPLE, SIMPLEC, and PISO at Steady State Conditions and Time Step Size of 1/100

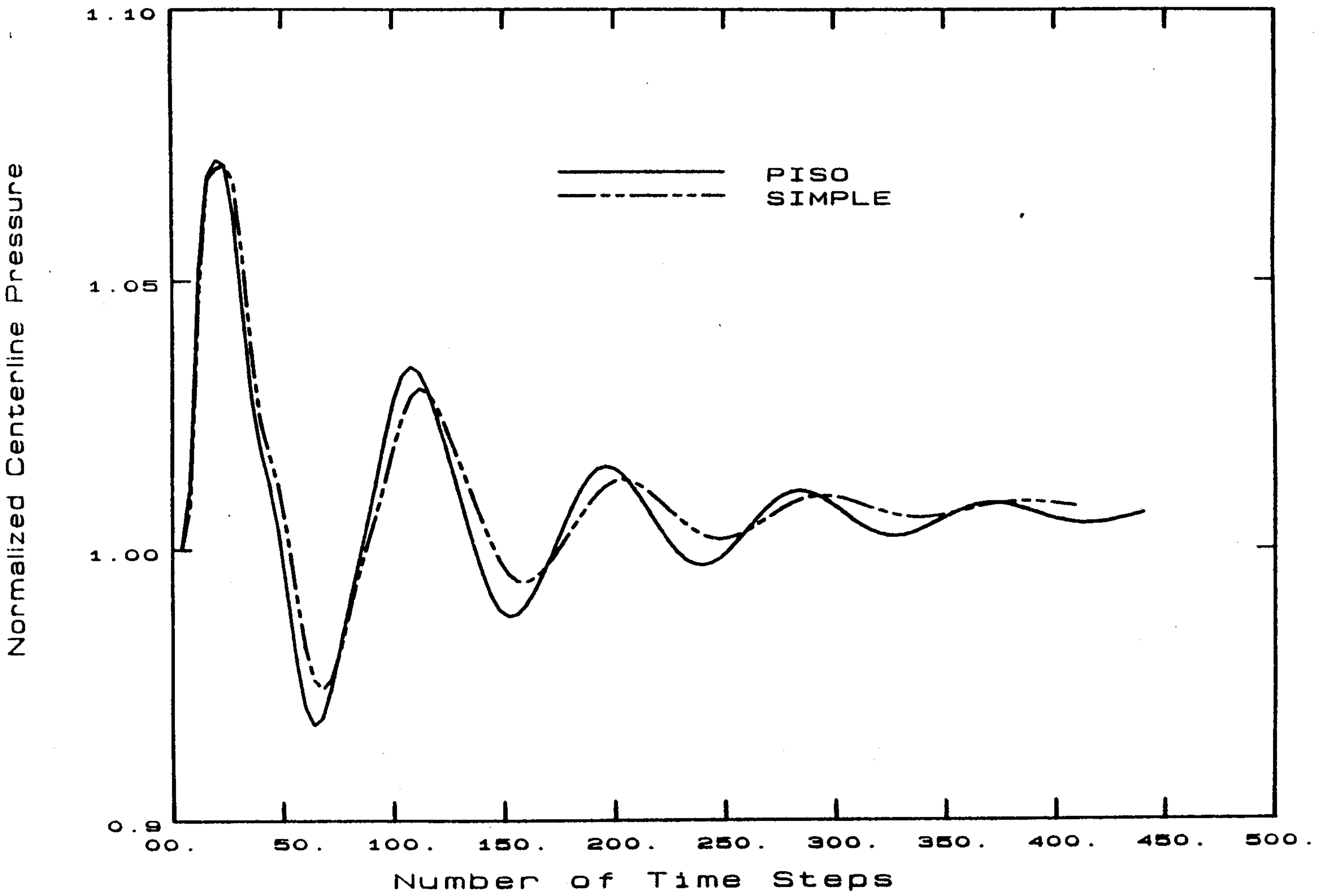


Figure 6.25 Comparison of PISO and SIMPLE Centerline Pressure for Case of 40x40 Grid Size and dt=1/100

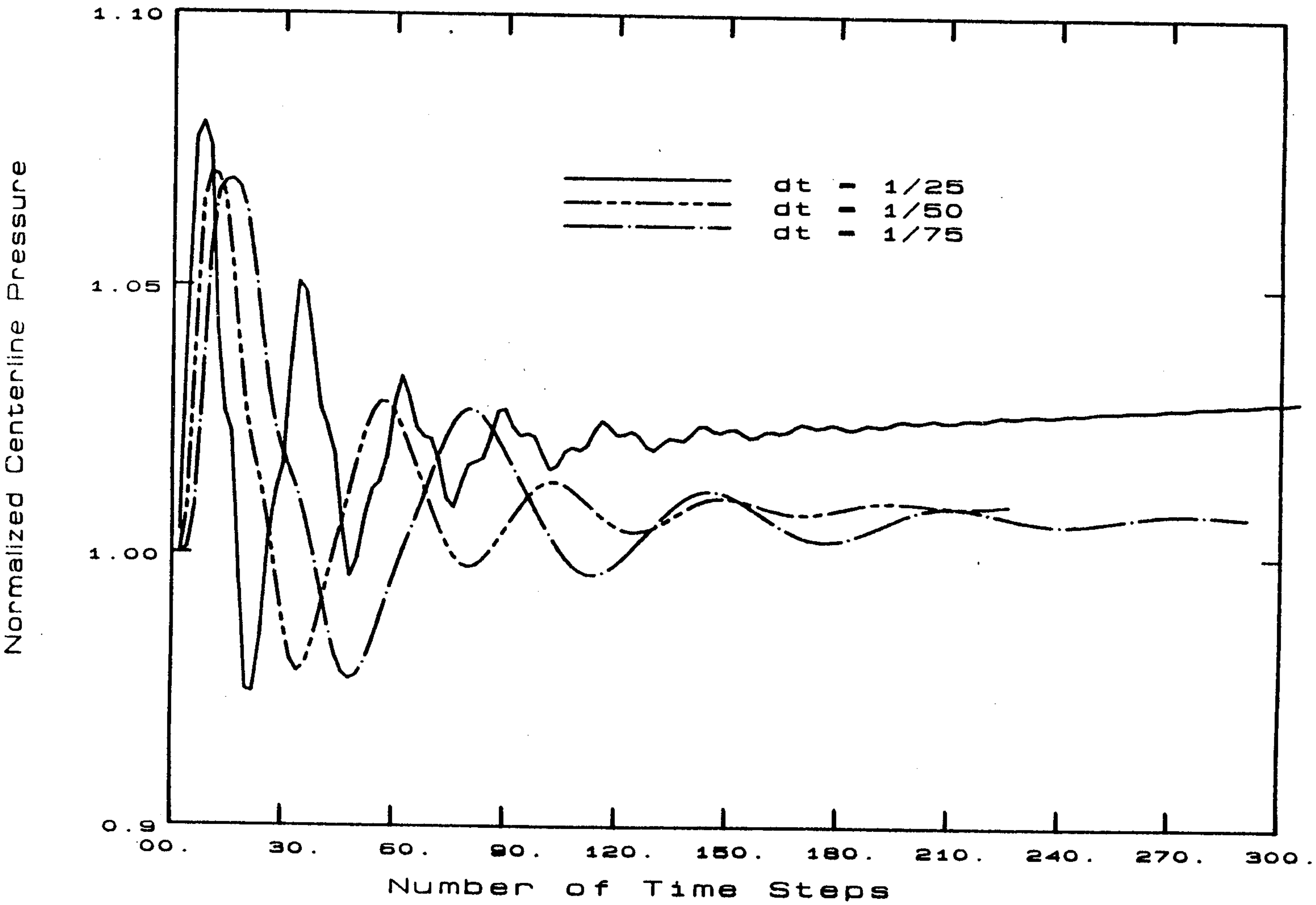


Figure 6.26 SIMPLE Centerline Pressure Variations for Case of 20x20 Grid Size

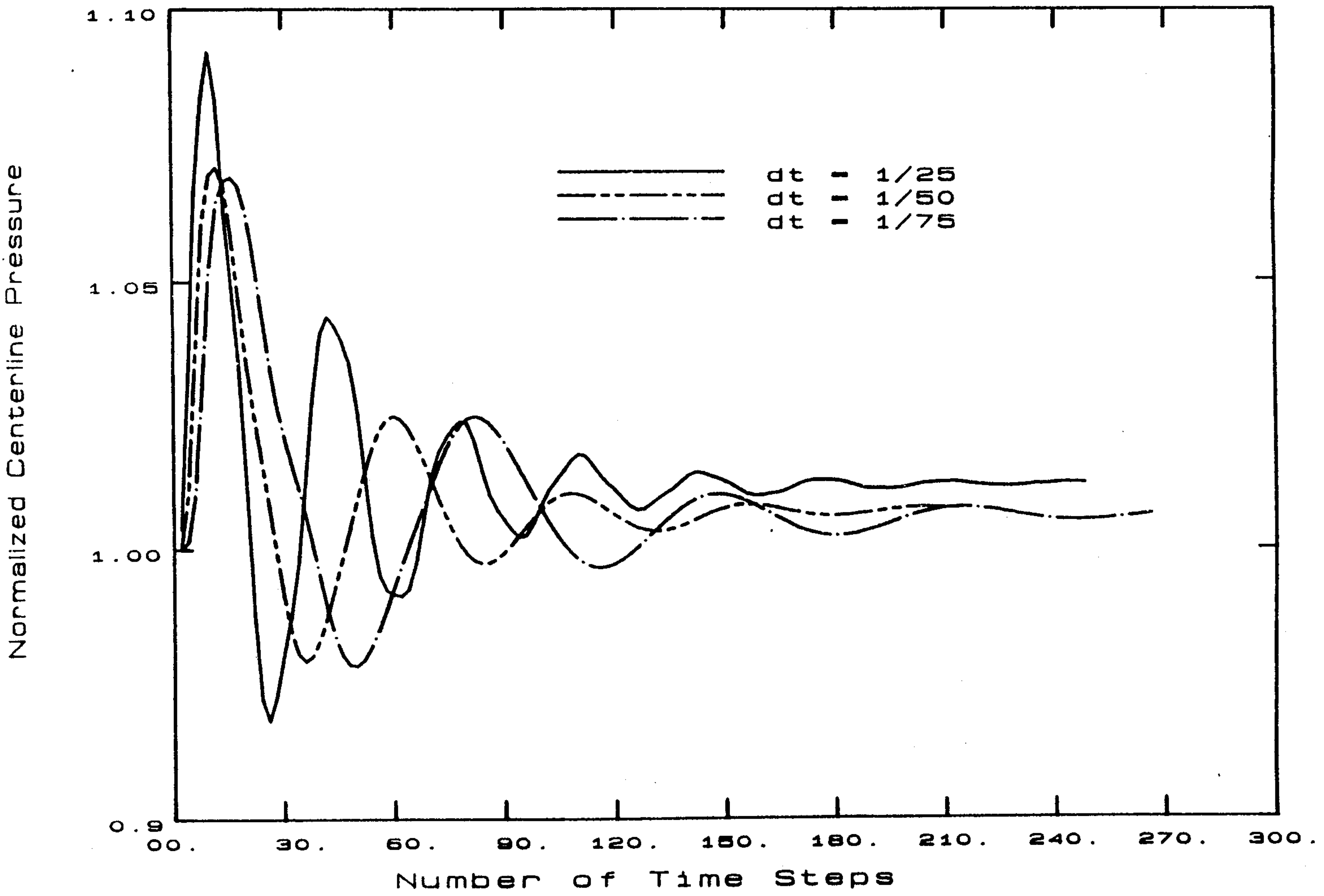


Figure 6.27 SIMPLEC Centerline Pressure Variations for Case of 20x20 Grid Size

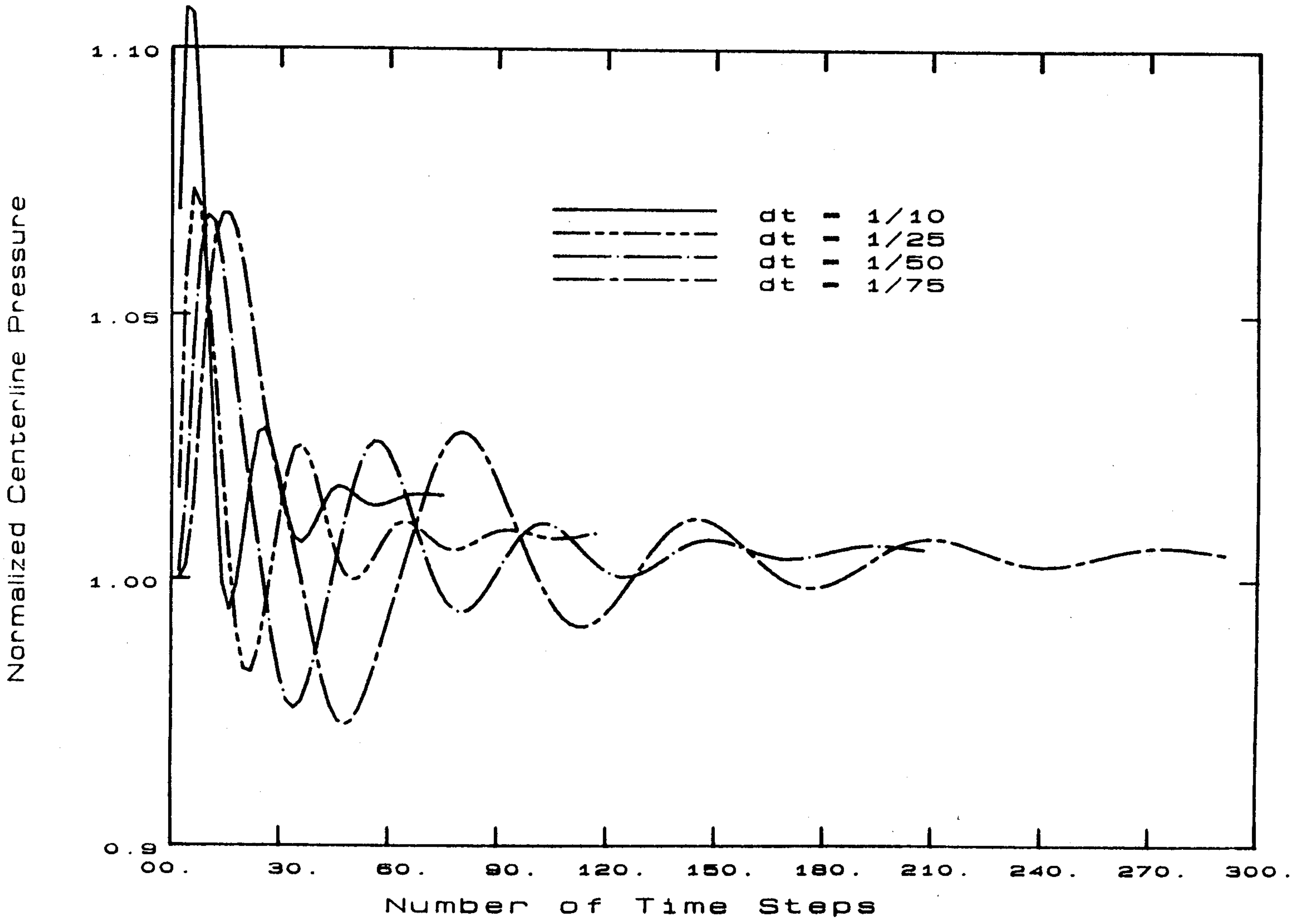


Figure 6.28 PISO Centerline Pressure Variations for Case of 20x20 Grid Size and Convergence Criterion of 1%

time steps per characteristic time period is equal to 25 or less. This observation raises the question of whether correct steady state values actually exist for this range of time step size values, even though the temporal numerical variations of the variables have decayed to negligible levels. The slight computational advantage that PISO held over the SIMPLE method as shown in Figure 6.20 may now be irrelevant if, indeed, the results produced in this time step size range are erroneous.

CHAPTER VII CONCLUDING REMARKS

In this study we have considered two pressure-velocity coupling schemes capable of handling both compressible and incompressible flow situations. The relative performance of each method was compared for two test cases involving laminar compressible flows in an axisymmetric sudden pipe expansion geometry.

Preliminary results obtained from the time-dependent compression/expansion test case indicated that pressure-temperature coupling effects are non-negligible and should be taken into account when a generalized approach is adopted for solving fluid flow equations. This study illustrates that the rigorous development of the PISO algorithm inherently and completely handles pressure-temperature coupling, whereas the SIMPLE algorithm requires additional density correction. While both algorithms yields results in agreement with the other, substantial time savings are realized in the calculation of time-dependent compressible flows by PISO over both the SIMPLE and SIMPLEC methods regardless of the time step size or grid size employed. Substantial numerical diffusion prevented a grid-independent solution to be obtained for either algorithm.

Results from the steady state test case again indicate good agreement between PISO, SIMPLE, and SIMPLEC numerical predictions. However, both SIMPLE and SIMPLEC display overall lower computational efforts than the PISO method when the solution procedure involves small temporal time step sizes. PISO does exhibit slight time savings over SIMPLE and SIMPLEC when larger time step sizes are employed, but the accuracy of the resulting field variables are questionable.

It is recommended that additional studies be carried out evaluating the performance of each algorithm under steady state, turbulent, compressible flow conditions. The strong coupling between the turbulence source terms and the velocity field may require enough under-relaxation in the SIMPLE algorithm such that the non-iterative PISO method would perform more effectively in terms of computational efficiency. An additional recommendation is that a higher order accurate discretization scheme be implemented in order to investigate the effects of reducing the numerical diffusion in the calculation domain where the velocity field is skewed to the coordinate grid.

List of References

Bai, S.D., 1986. "Numerical Simulation of Multiple Jet Interaction." Ph.D. Thesis, University of Alabama in Huntsville.

Bai, S.D., Wu, S.T., and Campbell, C.W., 1987. "Numerical Simulation of Single and Multiple Jet Interaction Within a Combustor." Proc. 5th International Conf. Numerical Methods in Laminar and Turbulent Flow, Montreal.

Benodekar, R.W., Goddard, A.J.H., Gosman, A.D., and Issa, R.I., 1985. "Numerical Prediction of Turbulent Flow over Surface-Mounted Ribs." AIAA J. 23, 359.

Bird, R.B., Stewart, W.E., and Lightfoot, E.N., 1960. Transport Phenomena. Wiley and Sons.

Briley, W.R., and McDonald, H., 1980. "On the Structure and Use of Linearized Block Implicit Schemes." J. Comput. Phys. 34, 54.

Ferziger, J.H., 1987. "Simulation of Incompressible Turbulent Flows." J. Comput. Phys. 69, 1.

Gosman, A.D., Pun, W.M., Runchal, A.K., Spalding, D.B., and Wolfshein, M., 1969. Heat and Mass Transfer in Recirculating Flows. Academic Press.

Gosman, A.D., and Watkins, A.P., 1977. "A Computer Prediction Method for Turbulent Flow and Heat Transfer in Piston/Cylinder Assemblies." Proc. 1st Symp. on Turbulent Shear Flows, 5.23, Pennsylvania.

Harlow, F.H., and Welch, J.E., 1965. "Numerical Calculation of Time-dependent Viscous Incompressible Flow of Fluid with Free Surface." Phys. Fluids 8, 2182.

Harlow, F.H., and Amsden, A.A., 1970. "The SMAC Method: A Numerical Technique for Calculating Incompressible Fluid Flows." Los Alamos Scientific Laboratory Rept. LA-4370.

Issa, R.I., 1985. "Solution of the Implicitly Discretised Fluid Flow Equations by Operator-Splitting." J. Comput. Phys. 62, 40.

Issa, R.I., Gosman, A.D., and Watkins, A.P., 1986. "The Computation of Compressible and Incompressible Recirculating Flows by a Non-iterative Implicit Scheme." J. Comput. Phys. 62, 66.

Kim, J. and Moin, P., 1980. J. Comput. Phys. 35, 31.

Leonard, B.P., 1979. "A Stable and Accurate Convective Modeling Procedure Based on Quadratic Upstream Interpolation." Comp. Meth. Appl. Mech. and Eng. 19, 59.

Milos, F.S., and Acrivos, A., 1987. "Steady flow past sudden expansions at large Reynolds number. II. Navier-Stokes solutions for the cascade expansion." Phys. Fluids 30, 7.

Patankar, S.V., and Spalding, D.B., 1972. "A Calculation Procedure for Heat, Mass, and Momentum Transfer in Three-dimensional Parabolic Flows." Int. J. Heat Mass Transfer 15, 1787.

Patankar, S.V., 1980. Numerical Heat Transfer and Fluid Flow. Hemisphere.

Pulliam, T.H., and Steger, J.L., 1980. "Implicit Finite-Difference Simulation of Three Dimensional Compressible Flow." AIAA J. 18, 159.

Raithby, G.D., 1976. "Skew Upstream Differencing for Problems Involving Fluid Flow." Comp. Meth. Appl. Mech. and Eng. 9, 153.

Sha, W.T., 1985. "Commix-1B: A Three Dimensional Transient Single-phase Computer Program for Thermal Hydraulic Analysis of Single and Multicomponent Systems." Argonne National Laboratory Rept. ANL-85-42, Vol. I.

Syed, S.A., Chiappetta, L.M., and Gosman, A.D., 1985. "Error Reduction Program - Final Report." NASA Contractor Report 174776.

Van Doormaal, J.P., and Raithby, G.D., 1984. "Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows." Numer. Heat Trans. 7, 147.

Van Doormaal, J.P., and Raithby, G.D., 1985. "An Evaluation of the Segregated Approach for Predicting Incompressible Fluid Flows." Presented at National Heat Transfer Conference, Denver, Colorado.

Van Doormaal, J.P., Raithby, G.D. and McDonald, B.H., 1987. "The Segregated Approach to Predicting Viscous Compressible Fluid Flows." J. Turbomachinery 109, 268.

Zedan, M., and Schneider, G.E., 1985. "A Coupled Strongly Implicit Procedure for Velocity and Pressure Computation in Fluid Flow Problems." Numer. Heat Trans. 8, 537.